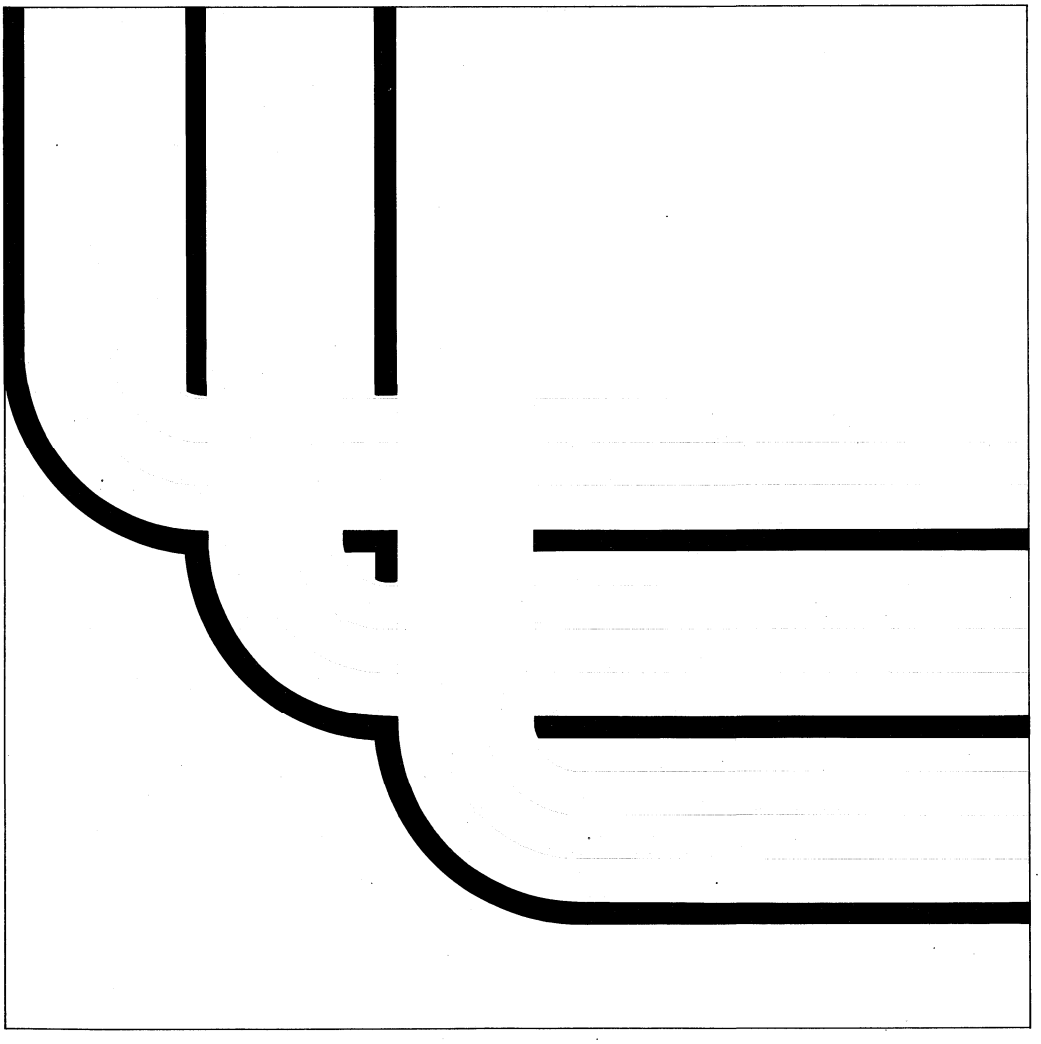
Application System/400™

SC41-8083-00

Security Concepts and Planning

Version 2



System and Application Support

Take Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page vii.

First Edition (April 1991)

This edition applies to the licensed program IBM Operating System/400, (Program 5738-SS1), Version 2 Release 1 Modification 0, and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the proper edition for the level of the product.

Order publications through your IBM representative or the IBM branch serving your locality. Publications are not stocked at the address given below.

A form for readers' comments is provided at the back of this publication. If the form has been removed, you may address your comments to:

Attn Department 245
IBM Corporation
3605 Highway 52 N
Rochester, MN 55901-7899

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you or restricting your use of it.

Contents

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM licensed program or other IBM product in this publication is not intended to state or imply that only IBM's program or other product may be used.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577.

The following terms, denoted by an asterisk (*) in this publication, are trademarks of the IBM Corporation in the United States and/or other countries:

Application System/400	AS/400
COBOL/400	IBM
NetView	Operating System/2
Operating System/400	OS/400
OfficeVision	RPG/400
Systems Application Architecture	System/370
SQL/400	400

The following terms, denoted by a double asterisk (**) in this publication, are trademarks of other companies as follows:

Century Schoolbook	American Type Foundry
ITC Avant Garde Gothic	International Typeface Corporation
ITC Souvenir	International Typeface Corporation
Monotype Arial	The Monotype Corporation plc.
Monotype Garamond	The Monotype Corporation plc.
Monotype Times New Roman	The Monotype Corporation plc.

This publication could contain technical inaccuracies or typographical errors.

This manual may refer to products that are announced but are not yet available.

Information that has changed since Version 1 Release 3 Modification 0 is indicated by a vertical bar (|) to the left of the change.

This publication contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

This manual contains small programs that are furnished by IBM as simple examples to provide an illustration. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. All programs contained herein are provided to you "AS IS". THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED.

Programming Interface

Appendix E, "Supported Call Level Interfaces" on page E-1 of this guide is intended help the customer set up security on the system-supplied application programming interfaces. It contains call level interfaces, their shipped default public authorities, their command equivalents (if any), and a brief description of their functions.

About This Manual

This manual provides information about security concepts and planning for the system. Security planning and implementation is the responsibility of the customer.

This manual helps you use the system security to:

- Protect your system and your data from being used by people who do not have the proper authorization
- Protect your data from intentional or unintentional damage or destruction
- Keep security information up to date
- Audit security-related events

This manual is intended for someone who is assigned the responsibilities of setting up users and controlling users' authorities on the system.

You should be familiar with the information contained in *System Operator's Guide*, SC41-8082, and *New User's Guide*, SC41-8211, before using this manual.

If you know how to operate the system, you should be ready to use this manual to prepare for your system security. To activate and maintain system security, use the security online help information on the system.

As you read the security information, boxes similar to the following are shown:

Security Risk or Consideration

These boxes contain information about important security risks or considerations.

This manual does not describe security for specific licensed programs, languages, and utilities.

Some of the information in this manual applies to Version 2, Release 1 Modification 1, which will be available at a later date. Version 2, Release 1, Modification 1 is also referred to as Version 2 Release 1.1.

You may need to refer to other IBM manuals for more specific information about a particular topic. The *Publications Guide*, GC41-9678, provides information on all the manuals in the AS/400 library.

For a list of related publications, see the "Bibliography."

Chapter 1. Introduction to Security

This chapter introduces security, a part of the Operating System/400* (OS/400*) licensed program. Security includes two types:

- Physical security
- Data Security

Physical security allows you to protect your system and other devices. **Data** security allows you to protect your data.

This chapter provides an overview of the controls that a security officer can use to limit access to the system and its data. More specific information is supplied later in this manual.

Security Overview

System security consists of the safeguards that you build into your system to help you achieve control over who can use your devices, data, and programs and to prevent accidental or intentional change or destruction of system resources.

Security helps prevent access to objects by users who do not have authority and helps protect the integrity of the data on the system. The term **object**, as used in this manual, is anything that exists in and occupies space in storage and on which operations can be performed, such as programs, files, libraries, and folders.

In many applications, controlling the integrity of the data is more critical than controlling the security of the data. To help achieve application integrity, you may want to ensure that:

- Changes to application data can only be made by users who should have the authority
- Changes to application data are made only through tested programs

Security Consideration

The security measures supplied by the AS/400* system can reduce the risk of users accidentally changing or destroying resources but does not prevent it. For your security measures to be effective, your resource controls should be combined with physical security and a division of duties. If these controls are not used, the system can be exposed to possible access by unauthorized people.

Security can help control:

- Access to the system by requiring the user to identify himself through the use of a user profile name (also known as a user ID) and a user password when signing on the system, or through the use of job descriptions for running batch jobs.
- Access to the system by requiring the remote user to identify himself through the use of a user profile name and a user password starting a communications job.

- Resources used by the system, such as display stations and printers, by verifying a users authority to use them.
- Data in the system by requiring that users have the authority to use specific objects such as files, programs, commands, and devices.
- Data in the system, on tape, or on diskette by storing data in encrypted format. The manual, *Cryptographic Support/400 User's Guide*, has more information about encrypting data.
- Functions on the system, such as adding users to the system, saving and restoring the operating system, performing service functions, and the control of other users' jobs, by requiring that users have special authority to use commands and programs.

The security controls provided by the operating system help you stop people from using system information for which they have no authority in the following ways:

- Setting the security level of the system to limit access to system resources
- Identifying users and verifying authority to the system by creating user profiles
- Authorizing users to resources by specifically giving users the authority to use those resources

Physical Security

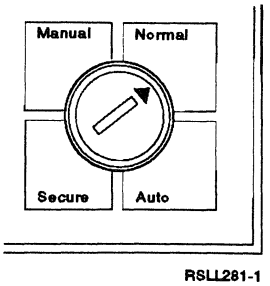
Physical security includes protecting the devices and media against damage and destruction and protecting the system from being used by people who do not have the authority. The following are a few of the things you should consider when deciding how to physically protect your system.

Protecting Your System

To help protect your system from damage or destruction, you should place your system in a room where there is less chance of it being damaged by fire or flooding. Placing your system in a locked room also helps prevent access to your system. This is an essential start at protecting your system.

Locking Your System Unit

You can limit the use of the whole system by using the keylock switch on the system control panel. This switch has four positions: Manual, Normal, Auto, and Secure.



RSLL281-1

The following table shows what operations can be performed for each keylock position:

Table 1-1. Keylock Positions

Operation	Manual	Normal	Auto	Secure
Manual load of the system	Yes	No	No	No
Automatic load of the system	Yes	Yes	No	No
Manual control of functions	Yes	No	No	No
Automatic operations such as remote initial program load	No	Yes	Yes	No
Powering down the system with the Power Down System (PWRDWNSYS) command from a display station	Yes	Yes	Yes	Yes
Turning the system off with the Power switch	Yes	No	No	No
Turning the system on with the Power switch	Yes	Yes	No	No
Do an initial program load of the system	Yes	No	No	No
Select a different load of the system	Yes	No	No	No
Perform dedicated service tools (DST) functions	Yes	No	No	No

Security Risk

Use the Manual position of the keylock switch only when loading the system, selecting a different initial program load, or using dedicated service tools. Some of the service tools allow a user to access all system resources.

In any of the four positions, the key can be removed from the keylock to keep the keylock in that position. Select the position you want and remove the key.

Security Consideration

To prevent users from manually turning off the system, it is recommended you turn the switch to the Normal position and remove the key.

You can also limit the use of one or more of your display stations with an optional lock on the display station. This lock prevents the use of a particular display station unless it is unlocked with a key.

For more information about operating your system, see the *Operator's Guide*.

Storing Your Tapes or Diskettes

Security Consideration

It is recommended that you save your system to tape on a regular basis and keep copies of the tapes at a different location in the event your working copy is damaged or destroyed. Diskettes may also be used for backing up selected objects.

To protect data on tape or diskette from unauthorized use or damage by fire, the media should be placed in a safe place such as a fireproof vault.

Data Security

Data security helps you prevent unauthorized people from signing on your system to get information on the system. Types of data security are:

- Password security
- Display station security
- Sign-on security
- Initial program or menu security
- Resource security

Security Levels

Your system is shipped without password and resource security active. This means that any user (including remote users starting communications jobs) can use any resource.

You can select the level of security for your system by changing the system value QSECURITY and then doing an initial program load (IPL) of the system. The level of security you select determines what default authority users are given. However, at all levels of security, you can change the system-supplied defaults and then specifically grant and revoke users' authorities for resources.

The following table compares the levels of security on the system.

Table 1-2. Security Levels

Function	Level 10	Level 20	Level 30	Level 40
User name required to sign on	Yes	Yes	Yes	Yes
Password required to sign on	No	Yes	Yes	Yes
Password security active	No	Yes	Yes	Yes
Resource security active	No	No	Yes	Yes
Menu and initial program security active	No	Yes[1]	Yes[1]	Yes[1]
Access to all objects	Yes	Yes	No	No
User profile created automatically	Yes	No	No	No
Record in auditing journal all programs that access objects using interfaces that are not supported	Yes	Yes	Yes	Yes
Record in auditing journal program instructions that are restricted from compiling	Yes	Yes	Yes	Yes
Programs that access objects using interfaces that are not supported fail at run time	No	No	No	Yes

Note:

[1] LMTCPB(*YES) must also be specified in the user profile.

Note: For more information about the auditing journal, see the topic "System-Provided Security Auditing Using Journals" on page 6-10.

User Profiles

User profiles are an important part of security. They are used to identify users to the system and verify users' authorities on the system. User profiles tell the system who can sign on and what functions the user can perform on the system resources after signing on. User profiles are created with the Create User Profile (CRTUSRPRF) command by the security officer or someone who has security administrator (*SECADM) special authority.

Security Consideration

Throughout this manual, the term **user** also means remote users starting communications jobs. Therefore, user profiles should be created for remote users to control what they can do on the system.

User profiles contain information that tailor the way a user operates on the system by using such things as:

- Special authority
- Display station security
- Sign-on security
- Initial program security
- Initial menu security
- Limited capability

For more information about user profiles, see Chapter 3, "User Profiles."

User Class

When identifying a user on the system, you can specify a user class in the user profile. The classification you specify determines what system control operations and what menu options he can use. The user classes are:

- Security officer (*SECOFR)
- Security administrator (*SECADM)
- Programmer (*PGMR)
- System operator (*SYSOPR)
- User (*USER)

Security Risk

Because the *SECOFR, *SECADM, *PGMR, and *SYSOPR user classes allow more control over system operations, you should make sure the user really needs the authority defined by the user class you specify.

Display Station Security

Display station security allows you to control who can sign on a display station. The system verifies authority to display stations to manage who can sign on.

If the security level is 10 or 20, it is possible for a user with *ALLOBJ or *SERVICE special authority to sign on any display station. If the security level is 30 or above, the default is that anyone can sign on to any display station with the following exceptions:

- The authority for the device restricts certain users.
- The system implicitly restricts users with *ALLOBJ or *SERVICE special authority.

  The only exception to this rule is the display station designated as the console. The security officer (QSECOFR), the service (QSRV), and the service basic (QSRVBAS) user profiles can always sign on the display station designated as the console even if they have not been given authority specifically to the device. However, if QSRV and QSRVBAS user profiles need to sign on a display station other than the one designated as the console, they must be given the required authority specifically to the device description.

You can allow a user with all object (*ALLOBJ) or service (*SERVICE) special authority to sign on to display stations other than the console by changing the limit security officer (QLMTSECOFR) system value.

Security Consideration

Be aware that before you change your system security level to 30 or above, you should give users with *ALLOBJ or *SERVICE the authority needed to specific device descriptions, or change the QLMTSECOFR system value to allow users who have *ALLOBJ or *SERVICE special authority to sign on any display station.

All users who do not have *ALLOBJ or *SERVICE special authority can sign on to any display station that has the public authority of *CHANGE specified in the device description.

For more information about display station security, see the topic "Display Station Considerations" on page 2-12.

Sign-On Security

Security Risk

Your system is shipped without password or resource security active (security level 10). Anyone can sign on the system by typing a user profile name from 1 to 10 characters in length. If a user profile by that name does not exist, the system creates one and the user is given the default authority for a system that does not have password and resource security active. The user profile name is also known as the user ID (user identification).

Sign-on security prevents a person who is not identified on the system from signing on. Sign-on security is active when your system security level is 20 or above. At level 20 or above, a user must specify a password in addition to a user ID (a user profile by that name must exist) to sign on the system. When the correct combination of user profile name and password is typed on the Sign On display, the person is allowed on the system.

For more information about specifying user names and passwords, see the topics "User Profile Name" on page 3-2 and "Password" on page 3-3.

The following is a sample Sign On display for security level 20 or above.

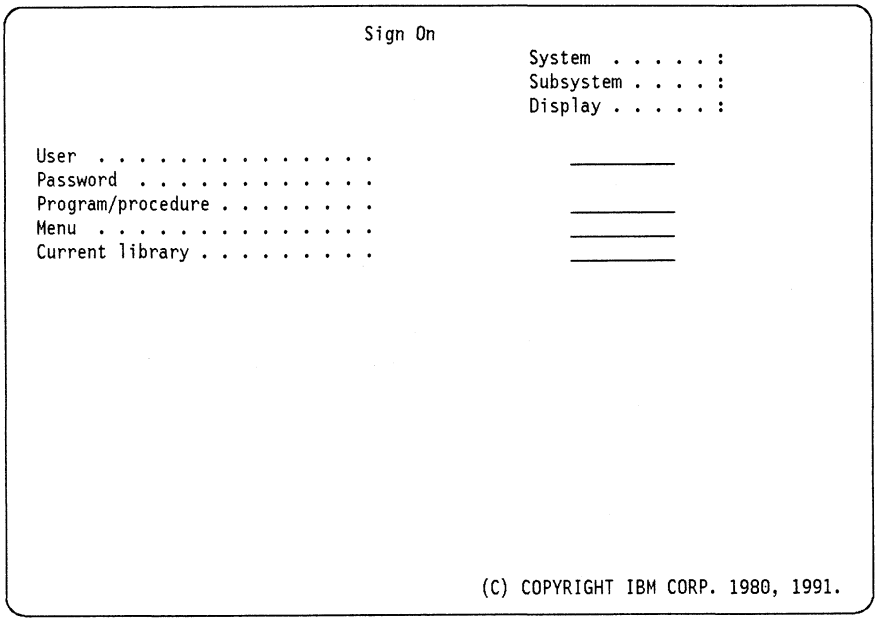


Figure 1-1. Sample Sign On Display for Security Levels 20 and Above

Notes:

1. Passwords are not shown when typed on this display.
2. The *Password* prompt is not shown on a level 10 system.

Initial Program and Menu Security

Initial program security allows you to specify a program to run when the user signs on the system. When initial program security is in effect, you can prevent the user from running a different program when he signs on. If you limit the user to a program, he cannot change the initial program value in his user profile or when he signs on.

For example, if you have a user whose only responsibility is to run an audit program once a month, you can prevent this user from running any other program by specifying the program is required when he signs on and then have the system sign him off after the program completes.

When using initial program security, you should specify that the user is a limited capability user (LMTCPB parameter).

For more information about initial program security, see the topic "Initial Program" on page 3-4.

Menu security is a good approach to use when all options on a menu control what a user can do. Menu security also allows you to prevent a user from displaying a different menu when he signs on. If you limit a user to a menu, he cannot change the initial menu value in his user profile. The default value in the user profile for the initial menu is the AS/400 Main Menu. What the user can do from this menu is controlled by his user class.

Security Consideration

When using menu security, you should specify that the user is a limited capability user (LMTCPB parameter).

For information about menu security considerations, see the topics "Initial Menu" on page 3-4 and "Menu Security" on page 5-12.

Limited Capability

You can limit a user's capability from specifying an initial program, an initial menu, a current library, or an Attention-key-handling program different than the one specified in his user profile.

Limiting a user's capability provides a way to control:

- Commands a user can enter on the command line of a menu
- The initial program that runs when the user signs on
- The initial menu shown after the user signs on
- The library where a user's newly created objects will be placed
- The Attention-key-handling program that runs when the user presses the Attn key

Limiting a user's capability is done in the user profile by specifying a *YES value on the limited capability (LMTCPB) parameter. For more information about the limited capability values, see "Limited Capability" on page 3-4.

Resource Security

Resource security is a way to ensure that only the people you give authority to can use certain resources such as files, libraries, and devices.

Security Consideration

When resource security (level 30 or above) is not active, anyone who signs on the system can use any file, library, or device on the system if special authorities specified in the user profile use the default values based on the system security level.

The following types of resource security allow you to control how a user can use a resource.

- System authority
- Library security
- Authorization lists
- Group profiles
- Adopted authority
- Authority holders

For more information on resource security, see Chapter 4, "Resource Security."

System Authority

The authority provided by the system allows you to tailor a user's environment on the system. The authorities on the system are divided into two major types:

Special authority
Specific authority

Special authority allows a user to perform system control operations that do not apply to specific objects such as saving the system, controlling other users' jobs, using the system service tools, controlling spooled files, and creating user profiles. Special authority is defined in the user profile by specifying the class of user or by tailoring the special authorities for a specific user. You use the Create User Profile (CRTUSRPRF) or the Change User Profile (CHGUSRPRF) command to define special authority. See the topic "Special Authority" on page 3-5 for more information.

Specific authority allows a user to perform specific operations on an object such as move or rename the object, control the object's existence, and perform operations on the data contained in the object. Specific authority is given to a user with the Grant Object Authority (GRTOBJAUT) command and removed using the Revoke Object Authority (RVKOBJAUT), or use the Edit Object Authority (EDTOBJAUT) command. See the topic "Specific Authority" on page 4-1 for more information.

The following figure summarizes the types of authority available on the system.

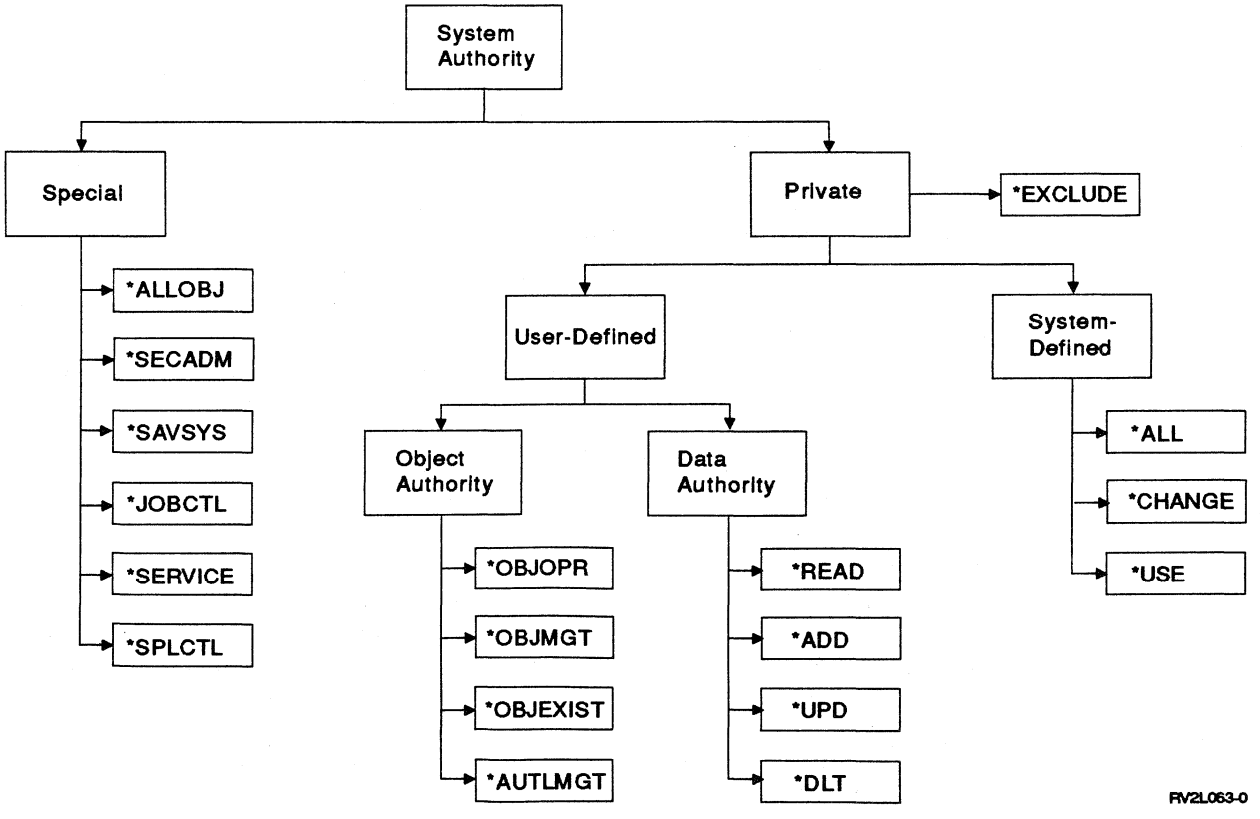


Figure 1-2. System Authorities

Exclude authority prevents a user from using an object. If exclude authority is specified, no other authority can be specified.

Security Risk

On a system without password security (level 10) active or on a system with only password security (level 20) active, all users have all object (*ALLOBJ) special authority and can use most functions or change any system resource.

Library Security

Library security allows you to place all sensitive objects in a library and then secure the library to limit the use of those objects. For example, if a payroll application is placed in a single library, you can then control security to this application by limiting authority to the library. Thus, you can prevent users from accessing any objects in the library by granting *EXCLUDE authority to the library. You can also use this approach to prevent a user from adding or deleting objects unless that user has specific authority to the library.

Security Consideration

If you give a user use (*USE) authority to the library, the user can still delete an object in the library if the user has object existence authority for that object in the library.

Additionally, you can combine the library security approach with specifically giving users authority. This allows you to give a group of users authority to the application but only give selected users in the group authority to certain files in the library.

You can also use the combination of the library security approach with specifically giving users authority to test libraries to protect production data while testing is taking place. For example, you may not want the development programmer changing production files and programs. You can assign a user to be responsible for controlling the production versions of the files and programs. This user can give the development programmer the test versions of the production files and programs. The development programmer can then prepare his source and test it with test data. After testing is complete, the development programmer gives the tested version of the files and programs to the user responsible for the production version. The user controlling the production version is then responsible for making the changes to the production library.

For more information about library security, see the topic "Default Public Authority for Newly-Created Objects" on page 4-22.

Authorization Lists

An authorization list is used to secure objects. An **authorization list** contains a list of users and the authority each user has to all objects that the authorization list secures. When you secure an object or a set of objects, you can specify an authorization list name instead of specifying authority for each user separately.

For more information about authorization lists, see the topic "Authorization Lists" on page 4-7.

Group Profiles

A **group profile** is a user profile used by a group of users. A group profile gives the same set of authorities to multiple users by allowing the member user profiles to use the authority of the group profile. A user profile that is a member of a group profile can do whatever the group profile has the authority to do as long as the member does not have any authority that overrides the authority of the group profile.

See "Authority Checking" on page 4-29 for more information about how authority is determined.

Security Risk

Care should be taken when specifying a user profile as a group such as the programmer (QPGMR) or the security officer (QSECOFR) user profile. Allowing a user to use QPGMR or QSECOFR user profiles as a group profile results in the group members having full job control such as canceling any job on the system.

For more information about group profiles, see the topic "Group Profiles" on page 4-9.

Adopted Authority

Adopted authority allows you to specify, when a program is created, that the program will always run under the program owner's user profile. A user does not need authority specifically given to him for the objects used by the program, but uses (adopts) the program owner's authority. The user has authority for the objects used by the program only when he is running the program and other programs called by the program.

Security Risk

Allowing a user to run a program adopting the owner's user profile is an intentional release of control that can allow unintentional access to objects.

For more information about adopted authority, see the topic "Programs That Adopt the Owner's Authority" on page 4-15.

Authority Holders

An **authority holder** allows you to specify the security information for program-described database files before the files actually exist on the system.

Security Risk

Authority holders are not recommended for systems that require closely controlled security. Authority holders are designed to provide a compatible function for System/36 applications running in the System/36 environment.

Then, when a file is created with the same name as the authority holder, the security information specified in the authority holder is linked to the file. This allows the system to keep authorities for System/36 environment applications that often delete files and then create them again. If the file is deleted, the authority information is kept with the authority holder to be used when the file is created again or restored from tape or diskette.

For more information about authority holders, see the topic "Authority Holders" on page 4-31.

Chapter 2. Security Considerations

This chapter discusses some aspects of security that the security officer should consider. These security considerations are:

- Security levels
- System values for security
- Display stations
- Authorization lists
- Group profiles
- Dedicated service tools
- Service user profiles
- Network attributes

Security Level Considerations

The system value QSECURITY is used to define the level of security on the system. Changing the QSECURITY system value does not immediately change system security. An initial program load (IPL) of the system must be performed before any changes in security occur. The system value QSECURITY can only be changed by a user profile that has all object (*ALLOBJ) and security administrator (*SECADM) special authorities.

Note: At all levels of security, the system will do the following if the auditing journal QAUDJRN has been created and the system value QAUDLVL has been set:

- Write a journal entry for all attempts to access objects, using interfaces that are not supported, to the auditing journal QAUDJRN.

  This includes attempts to directly call system programs not documented at the call level interfaces, or attempts to access internal system structures through pointer capabilities of languages such as C, PASCAL, assemblers, and so forth. For example, directly calling the command processing program for the SIGNOFF command would cause a journal entry.

  See Appendix E, "Supported Call Level Interfaces" on page E-1 for a list of call-level interfaces that are supported.

- Write a journal entry for all programs that use restricted instructions each time they are run to the auditing journal QAUDJRN (if it exists). Security auditing must be requested using the QAUDLVL system value.

- Restrict programs that contain restricted instructions from compiling.

  Note: This is restricted at all security levels, regardless of whether QAUDLVL is set on or if QAUDJRN is created.

For more information about creating the auditing journal QAUDJRN and setting the QAUDLVL system value, see the topic "Setting Up Security Auditing" on page 6-11.

For more information about changing the QSECURITY value, see the topic "Changing the System Security Level" on page 7-13. For information about the

default special authority values for the different security levels, see Table 2-1 on page 2-2 and Table 2-2 on page 2-4.

Security Level 10 Considerations

Your system is shipped with minimal security active and does not require a password to sign on. If a user profile does not exist with the same name as the user ID that the user signs on with, the system creates a user profile with that name.

Security Risk

Security level 10 is a security risk because, by default, the system gives the user authority to access or delete any object on the system.

However, at this level you can tailor a user's special authorities. If you specify a user class in the user's profile and remove or add special authorities, the special authorities you specify in the SPCAUT parameter will override the defaults defined for the user class. You can grant and revoke users' authorities to see how security works before changing the security level to 30 or above.

The limited capability value does **not** apply to security level 10. For example, if a user profile is created with LMTCPB(*YES), the user is not prevented from changing his initial menu, initial program, or current library values. The user can run commands, from the command line of a menu, that can be restricted on a system with security level of 20 or above.

The following table shows the default special authorities given to each user when creating or changing a user profile if the special authority value in the user profile is specified as user class, SPCAUT(*USRCLS).

Table 2-1. Special Authority for Level 10 or 20

User Class	*ALLOBJ	*SECADM	*SAVSYS	*JOBCTL	*SERVICE	*SPLCTL
*SECOFR	X	X	X	X	X	X
*SECADM	X	X	X	X		
*PGMR	X		X	X		
*SYSOPR	X		X	X		
*USER	X		X			

When changing from a security level 30 or above to a security level 20 or below, all user profiles have special authorities added or removed based on the user class (USRCLS). At security level 20 or below, the default (*ALLOBJ) special authority allows all users access to all system resources.

Security Level 20 Considerations

Minimal security is active and a password is required to sign on. A user profile must already exist for a user before he can sign on the system.

Security Risk

Security level 20 is a security risk because, by default, the system gives the user authority to access or delete any object on the system after he signs on.

However, at this level you can tailor a user's special authorities. If you specify a user class in a user profile and also remove or add special authorities, the special authorities you specify in the SPCAUT parameter will override the defaults defined for the user class. You can grant and revoke users' authorities at this security level to see how security works before changing the system security level to 30 or above.

The limited capability value specified in the user profile applies at security level 20. For example, if a user profile is created with LMTCPB(*YES), the user is prevented from changing his initial menu, initial program, or current library values. The user can run only commands with the allow limited user (ALWLMTUSR) parameter specified as *YES from the command line of a menu. The commands created by IBM* with the ALWLMTUSR parameter specified as *YES are: Sign Off (SIGNOFF), Send Message (SNDMSG), Display Message (DSPMSG), Display Job (DSPJOB), Start PC Organizer (STRPCO), and Display Job Log (DSPJOBLOG).

To allow a limited capability user to use commands other than the ones mentioned previously, you can use the Change Command (CHGCMD) command and specify *YES or *NO on the ALWLMTUSR parameter.

The default special authorities given for each user when creating or changing a user profile are the same as for level 10. See Table 2-1 on page 2-2.

When changing from a security level 30 or above to a security level 20 or below, all user profiles have special authorities added or removed based on the user class. This allows the user to access all system resources.

Security Level 30 Considerations

Password security is active and users must be specifically given authority to resources (resource security). A user profile must exist for the user, and a password is required to sign on.

When the system security level is 30, the system writes a journal entry to the auditing journal for the following, if the auditing journal QAUDJRN exists and the system value QAUDLVL has been set:

- Users submit jobs using a job description containing a user profile name and the requesting user does not have *USE authority to that user profile
- Users sign on without entering a user ID or password. The work station entry for the user references a job description that has a user profile name specified for the USER parameter, which allows the user to sign on by pressing the Enter key.

Security Consideration

Security level 30 or above is recommended because the system does not give the user authority to access all objects on the system after he signs on.

Although security level 30 provides more protection than security levels 10 or 20, security level 40 provides greater protection for businesses that have strong security requirements (see "Security Level 40 Considerations" on page 2-5 for more information about security level 40).

The limited capability value specified in the user profile applies at security level 30. For example, if a user profile is created with LMTCPB(*YES), the user is prevented from changing his initial menu, initial program, or current library values. The user can run only commands with the allow limited user (ALWLMTUSR) parameter specified as *YES from the command line of a menu. The commands created by IBM* with the ALWLMTUSR parameter specified as *YES are: Sign-off (SIGNOFF), Send Message (SNDMSG), Display Message (DSPMSG), Display Job (DSPJOB), Display Job Log (DSPJOBLOG), Start PC Organizer (STRPCO), and any command with the allow limited user parameter specified as ALWLMTUSR(*YES).

To allow a limited capability user to use commands other than the ones mentioned previously, you can use the Change Command (CHGCMD) command and specify *YES or *NO on the allow limited capability user (ALWLMTUSR) parameter.

The Check Limited Capability (CHKLMTCPB) command in library QUSRTOOL provides a simple method of determining which users with user class *USER have the LMTCPB parameter specified as *NO.

The following figure shows the default special authorities given for users when creating or changing a user profile if the special authority value is specified as user class, SPCAUT(*USRCLS).

Table 2-2. Special Authority for Level 30 or above

User Class	*ALLOBJ	*SECADM	*SAVSYS	*JOBCTL	*SERVICE	*SPLCTL
*SECOFR	X	X	X	X	X	X
*SECADM		X	X	X		
*PGMR			X	X		
*SYSOPR			X	X		
*USER	No	special	authority			

When changing from a security level 10 or 20 to a security level 30 or above, all user profiles have special authorities added or removed based on the user class (USRCLS). A user class of *USER does not have any special authority. Only a user class of *SECOFR has spool control (*SPLCTL) special authority and service (*SERVICE) special authority.

Security Level 40 Considerations

Password security, resource security, and operating system integrity are active. Users must be specifically given authority to resources (resource security). A user profile must exist for the user, and a password is required to sign on. All security level considerations for security level 30 also apply to security level 40. When changing from security level 30 to security level 40, special authorities set by the user class are not changed.

Security level 40 provides more protection than security level 30.

- All attempts to access objects using interfaces that are not supported fail.

  This includes attempts to directly call system programs not documented as the call-level interfaces, or attempts to access internal system structures through pointer capabilities of languages such as C/400, Pascal, assembler, and so forth. For example, directly calling the command processing program for the SIGNOFF command will fail and cause a journal entry to be written to the QAUDJRN security journal.

  See Appendix E, "Supported Call Level Interfaces" for a list of call-level interfaces that are supported.

- Programs that contain restricted instructions will not compile.
- Users submitting jobs using a job description containing a user profile name must have *USE authority to that user profile.
- Users signing on without entering a user ID or password will fail. If the work station entry for the user references a job description that has a user profile name specified for the USER parameter, then the user can sign-on by pressing the Enter key.
- The system checks for a validation value when restoring programs to the system. If the program does not have a validation value, or if the validation fails, the program is translated again. See "Limiting the Restore of Programs That are Not Valid or Were Changed" on page 7-10 for more information about the validation value.

  If the translation fails (no template exists, a restricted instruction is used), the program is restored, but all public and private authority is revoked and the ownership is transferred to QDFTOWN. When the validation value does not exist and the translation fails, or if the validation of the value fails, the information is recorded in the auditing journal, in the job log as a message, and in the restore report (if one is requested).

  Translation is also controlled by the allow object differences (ALWOBJDIF) parameter on the restore commands. If ALWOBJDIF(*ALL) is specified, and a validation value does not exist, no translation is done.

  For more information about the validation value, see the topic "Limiting the Restore of Programs That are Not Valid or Were Changed" on page 7-10.

- Enhanced hardware storage protection.

  At security level 40 system control blocks located on disk are protected from modification. You can read the control blocks, but you cannot write to them. This support does not apply at security levels less than 40.

The following table identifies which AS/400 system models support enhanced hardware storage protection.

Table 2-3. AS/400 System Models With Enhanced Hardware Storage Protection

AS/400 System Model	Enhanced Hardware Storage Protection Provided
9402 D06	Not supported
9402 D04	Not supported
9404 D10	Not supported
9404 D20	Not supported
9404 D25	Supported
9406 D35	Supported
9406 D45	Supported
9406 D50	Supported
9406 D60	Supported
9406 D70	Supported
9406 D80	Supported

Security Consideration

Security level 40 is recommended for new systems **after** your system has run at security level 30. You should monitor the auditing journal at level 30 or below for violations and change any programs that have errors before you operate your critical programs at security level 40.

See Chapter 6, "Auditing Security for the AS/400 System" on page 6-1 for more information about monitoring your system.

Considerations for Migrating to Security Level 40

When the security level is changed to 40, protection from access by using interfaces that are not supported becomes active. The protection includes prevention of direct calls to system programs by user programs or by changed objects other than through supported interfaces.

Before going to security level 40, the auditing journal (QAUDJRN) should be created and the system value controlling security auditing (QAUDLVL) must be set to include authority failures (*AUTFAIL) and program failures (*PGMFAIL). All critical applications should be run and the auditing journal checked for authority failure violations. Each journal entry for an authority failure represents a program failure when security level 40 is active.

Because checking is done on programs during restore operations, backup and recovery procedures should be reviewed and changes should be made before security level 40 is activated.

Programs that were created on a release prior to 3.0 that have not been translated on Release 3 or later, or have not been created again using the FRCCRT parameter on the Change Program (CHGPGM) command, will not contain the required validation value as described in "Limiting the Restore of Programs That are Not Valid or Were Changed" on page 7-10. When the programs are restored to a system that has security level 40 active, the system will attempt to create them again. This can add a considerable amount of time to the restore operation. If the system is not able to create the programs again, the restored copy

has public and private authorities revoked and the ownership of the program is changed to QDFTOWN user profile.

For these reasons, it is recommended that you use the CHGPGM command to force the program to be created again before security level 40 is active. Once the program has be created again successfully on release 3 or later, the additional restore time is avoided.

Forcing the program to be created again may be advisable if there are any concerns about where the program came from or what the program is used for.

Specifying ALWOBJDIF(*ALL) on the restore command can prevent the system from attempting to create the program again at security level 40. However, this parameter should be used only when the content of the media is trusted and known, and when other factors controlled by this parameter are appropriate.

If programs have the observable information deleted or contain restricted instructions that cannot be created again by CHGPGM, the appropriate CRTxxxPGM command must be used.

If the system is running at security level 30 or below, the following procedure is recommended:

1. Run at security level 30.
2. Activate the auditing journal by creating the journal receiver using the Create Journal Receiver (CRTJRNRCV) command and the QAUDJRN journal using the Create Journal (CRTJRN) command.
3. Specify *PGMFAIL for the system value QAUDLVL.
4. Monitor the auditing journal at security level 30 for violations while running normal applications.
5. Change any programs that have errors using the Change Program (CHGPGM) command to translate all application programs again. This will not correct any errors found in the QAUDJRN journal.
6. Change the QSECURITY system value to 40 and do an IPL of the system.

Security Risk

Auditing at security level 30 detects almost all situations that would fail at security level 40. Hardware storage protection violations can only be detected at security level 40 and will not be detected while auditing at lower security levels.

Among the internal system control blocks protected by hardware storage protection are the System Entry Point Table and Work Control Blocks.

Considerations for System Values and Network Job Values

A number of system values allow you to tailor the security for the system. These values control sign on, passwords, inactive jobs, and network jobs.

Security-Related System Values

Table 2-4 shows the system values related to security. The following values can be changed by the security officer (QSECOFR) user profile or a user profile that has all object (*ALLOBJ) and security administrator (*SECADM) special authorities.

For more information about the system values and the values that can be specified, see the topic "Changing the System Security Level" on page 7-13.

Table 2-4. System Values Related to Security

Value	Description
QAUDLVL	Determines the level of auditing on the system.
QAUTOVRT	Determines if virtual device descriptions are created automatically.
QDSPSGNINF	Determines if the Sign On display is shown when a user signs on.
QINACTITV	Determines the interval in minutes that a work station is inactive before a message is sent to a message queue, or that the job at the work station is automatically ended.
QINACTMSGQ	Specifies the name of the message queue that receives messages about the work station when the inactive time interval is reached, or specifies that the job at the work station will automatically end when the interval is reached.
QLMTDEVSSN	Determines if a user can have more than one device session occurring at one time.
QLMTSECOFR	Determines if users with *ALLOBJ or *SERVICE special authorities can sign on to devices.
QMAXSIGN	Determines the maximum number of sign-on attempts that are not valid.
QMAXSGNACN	Determines the action taken by the system when the QMAXSIGN limit is reached. The device, user profile or both can be disabled.
QPWDEXPITV	Determines the maximum number of days that a password is valid.
QRMTSIGN	Determines if automatic sign-on from a remote system is allowed.
QSECURITY	Determines the system security level.

Table 2-5 on page 2-9 shows the system values that apply to passwords when the Change Password (CHGPWD) command is used. These values do not apply to the Change User Profile (CHGUSRPRF) and Create User Profile (CRTUSRPRF) commands.

Table 2-5. System Values That Apply to the Change Password Command	
Value	**Description**
QPWDLMTAJC	Determines if digits can be next to each other in a new password.
QPWDLMTCHR	Determines the characters that cannot be used in a new password.
QPWDLMTREP	Determines if repeating characters can be used in a new password.
QPWDMINLEN	Determines the minimum number of characters in a password.
QPWDMAXLEN	Determines the maximum number of characters in a password.
QPWDPOSDIF	Determines if each position in a new password must be different from the old password.
QPWDRQDDGT	Determines if a digit is required in a new password.
QPWDRQDDIF	Determines if the password must be different from the 32 previous passwords.
QPWDVLDPGM	Specifies the name of the user-written password approval program.

For a description of the values that can be specified for system values related to security, see the topic "Working with System Values That Affect Security" on page 8-10.

The *Work Management Guide* has more information about system values. To change security-related system values, you must have *ALLOBJ and *SECADM special authorities.

Security-Related Network Attributes

The following topics explain the network attributes that are related to security. These attributes control the way the AS/400 system processes incoming requests from remote systems, personal computers, and distributed services.

Job Action Considerations for Object Distribution

The network attribute JOBACN is used to determine how the AS/400 system processes incoming requests from a remote system. When the AS/400 system receives incoming input streams as a target system, there are three different input stream options that can be used. The system can be set to reject (*REJECT) all incoming input streams, file (*FILE) incoming input streams, or base it on the network job table (*SEARCH). The network attribute JOBACN is initially set to *FILE.

***REJECT:** The input stream is rejected. A message stating the input stream was rejected is sent to both the sender and the intended receiver.

***FILE:** The input stream is filed on the queue of network files for the receiving user. This user can display, cancel, or receive the job stream into a database file or submit it to a job queue. A message stating that the input stream was filed is sent to both the sender and the receiver.

SEARCH: The network job table controls the actions by using the values in the table.

For more information about the JOBACN attribute, refer to the *Distribution Services Network Guide*.

AS/400 PC Support Access Considerations

All AS/400 PC Support users must be enrolled in the system distribution directory to obtain changes to PC Support programs. To enroll a user in the system distribution directory, the user must have security administrator (*SECADM) special authority.

PCSACC (PC Support Access) Parameter

The network attribute PCSACC is used to determine how the AS/400 system processes requests from attached personal computers. This value is initially set to *OBJAUT.

Three options control access. The system can reject PC Support requests to prevent access, can use object authorization to determine which users can access data, or can combine object authorization with a user-written exit program to further restrict getting PC Support requests.

REJECT: Specifies that the system does not allow any PC Support requests from remote systems. However, the system (as a local system) can still use PC Support to access other systems that allow it. A system cannot access files on any AS/400 system that specifies *REJECT.

OBJAUT: Specifies that all PC Support requests are allowed, but they are controlled by the object authorizations on this system.

qualified-program-name: Specifies the name of the user-written exit program (and the library in which it is stored) that provides additional security to the AS/400 object-level security (which still applies). This user-written exit program is passed by a parameter list, built by the remote system, that identifies the local system user and the request. The program is used to determine whether to allow the request.

User-Written Exit Program: AS/400 PC Support allows user-written exit programs. With an exit program, you can determine if the requester of a PC Support function should be allowed to perform the function and what data he is allowed to access. To start a user-written exit program that will run each time PC Support is used on the AS/400 system, you need to run the Change Network Attributes (CHGNETA) command to change the PCSACC parameter. The format for this command is:

```
CHGNETA PCSACC(library-name/program-name)
```

A user-written exit program becomes effective for function requests using PC Support after the command is run. Then, for every requested function, the system calls the specified program passing two parameters to it: a return code area and a character data structure containing the user profile name and the request. The information in the character data structure can be analyzed by the program to determine if the request should be allowed.

If the program sets the return code to hex F1, the request is allowed. If the program sets the return code to something other than hex F1, the function is

rejected. A message is sent to the requesting personal computer, stating the requested function is rejected.

The fields in the character data structure, except for the user profile name, are different for the virtual printer function, message function, and transfer function.

You can find a description of the data structure and an example of a user-written exit program in the manual *PC Support/400 Technical Reference for DOS and OS/2*.

Note: To control PC Support shared folders, see the topic "Distributed Data Management Considerations."

Distributed Data Management Considerations

The network attribute DDMACC is used to determine how the AS/400 system, as a remote system, processes requests from other systems. This parameter is initially set to *OBJAUT.

Shared folders cannot be controlled by a PC Support user-written exit program. Shared folders can be controlled using a distributed data management (DDM) exit program written by a user. If you plan to limit the right to create folders on the AS/400 system, be aware that, unless you write an exit program, a PC Support user will be able to create an unlimited amount of folders by assigning a drive and using the Make Directory (MD) command.

When the AS/400 system is a remote system, three options control access to its database files. The system can reject DDM requests to prevent access to its database files, can use object authorization support to determine which users can access what files, or can combine object authorization support with a user-written exit program to further restrict accessing a file.

***REJECT:** Specifies that the system does not allow any DDM requests from remote systems. However, the system (as a local system) can use DDM to access files on the other systems that allow it. A system cannot access files on any AS/400 system that specifies *REJECT.

***OBJAUT:** Specifies that all remote requests are allowed, but they are controlled by the object authorizations on this system.

***qualified-program-name:** Specifies the name of the user exit-written program (and the library in which it is stored) that provides additional security to AS/400 object-level security (which still applies). This user-written exit program is passed a parameter list, built by the remote system, that identifies the local system user and the request. The program is used to determine whether to allow the request.

For more information about the DDMACC attribute and exit program, refer to the *DDM Guide*.

Subsystem Considerations

Care must be taken when defining subsystems (object type *SBSD) to make sure that users are correctly identified on the system. To make sure users are correctly identified:

1. A user ID and password must be entered by all users.

Do not create a job description that will be used in a work station entry command with the following specifications:

```
CRTJOBD  JOBD(USERJOBD) USER(QUSER)
ADDWSE   SBSD(SAMPLE) WRKSTNTYPE(*ALL) JOBD(USERJOBD)
```

Security Consideration

In the example, the job description USERJOBD is created with the user profile name QUSER. Therefore, the user does not need to type a user ID and password. If the user presses the Enter key on the Sign On display, the user is signed on using the QUSER user profile.

2. All communications jobs must have a user ID and password.

   Do not add a communications entry with the following specifications:

   ```
   ADDxxxCME   SBSD(SAMPLE) DEV(*ALL) DFTUSER(QUSER)
   ```

   Security Consideration

   In the example, the default user parameter specified as DFTUSER(QUSER) allows the system to accept job start requests without a user ID or password from a communications request. The communications job is signed on using the QUSER user profile.

Display Station Considerations

Security Risk

If the security level is 10 or 20, anyone can sign on to any display station.

When the security level is 30 or above, the default is that anyone can sign on to any display station with the following exceptions:

- The authority for the device restricts certain users.
- The system implicitly restricts users with *ALLOBJ or *SERVICE special authority.

The only exception to this rule is the display station designated as the console. The security officer (QSECOFR), service (QSRV), and service basic (QSRVBAS) user profiles can always sign on the display station designated as the console, even if they have not been given authority specifically to the device. System value QCONSOLE is checked to determine the console device.

Security Consideration

Be aware that before you change your security level to 30 or above, you should give users with *ALLOBJ or *SERVICE special authority change (*CHANGE) authority to a device description, or change the limit security officer (QLMTSECOFR) system value to give these users the ability to sign on. If you do not give them *CHANGE authority to a device description or you do not change the QLMTSECOFR system value, they cannot sign on after the security level has been changed. Authority is given to a device description using the Edit Object Authority (EDTOBJAUT) or the Grant Object Authority (GRTOBJAUT) command.

Change the system value QLMTSECOFR to 0 (zero) to allow all users with *ALLOBJ or *SERVICE special authority to sign on any display station.

All other users can sign on to any display station that has the public authority of *CHANGE specified for the device description, unless the user has a private authority of less than *CHANGE to the device description.

For example, assume user profile USER1 is a member of a group profile GROUP1. If USER1 or GROUP1 does not have all object (*ALLOBJ) or service (*SERVICE) special authority but does have change (*CHANGE) authority to the display station, then USER1 is allowed to sign on.

If USER1 and GROUP1 do not have *CHANGE or less authority to the display station but user *PUBLIC has *CHANGE authority, then USER1 is allowed to sign on. However, if USER1, GROUP1, or *PUBLIC does not have *CHANGE authority, then the sign-on fails, and a message is sent telling the user he does not have authority for the display station.

In contrast, user profiles or group profiles that have all object (*ALLOBJ) or service (*SERVICE) special authority must have change (*CHANGE) authority given specifically for the display station, the QLMTSECOFR system value must be set to 0 (zero), or the QSECOFR user profile must have *CHANGE authority to the device description. See Figure 2-1 on page 2-15 for details.

For example, assume USER2 is a member of a group profile GROUP2 and GROUP2 has *ALLOBJ special authority. For USER2 to sign on the display station, USER2, GROUP2, or QSECOFR must have *CHANGE authority given specifically to them for the display station or the QLMTSECOFR system value must be set to 0 (zero). Any authority specified for user *PUBLIC for the display station is ignored.

Security Consideration

For systems that require maximum security, user profiles with all object (*ALLOBJ) or service (*SERVICE) special authority should be limited to the console. Normally, you would not want the security officer or service user profiles to sign on from a remote location. To prevent this, the system requires these user profiles to be properly authorized. To prevent the security officer (QSECOFR), QSRV, or QSRVBAS user profile from signing on to a display station other than the console, the QLMTSECOFR system value should be set to 1 (one). The device descriptions should be created using the programmer (QPGMR) user profile.

Authority Checking: The following figure shows the order that the system verifies authority for a display station. The basic rule is that *CHANGE authority is required for a user to sign on a display station. If a user signing on the system has a private authority to the device description, then the checking of authority to the device stops and this private authority is used to determine sufficient access to the system.

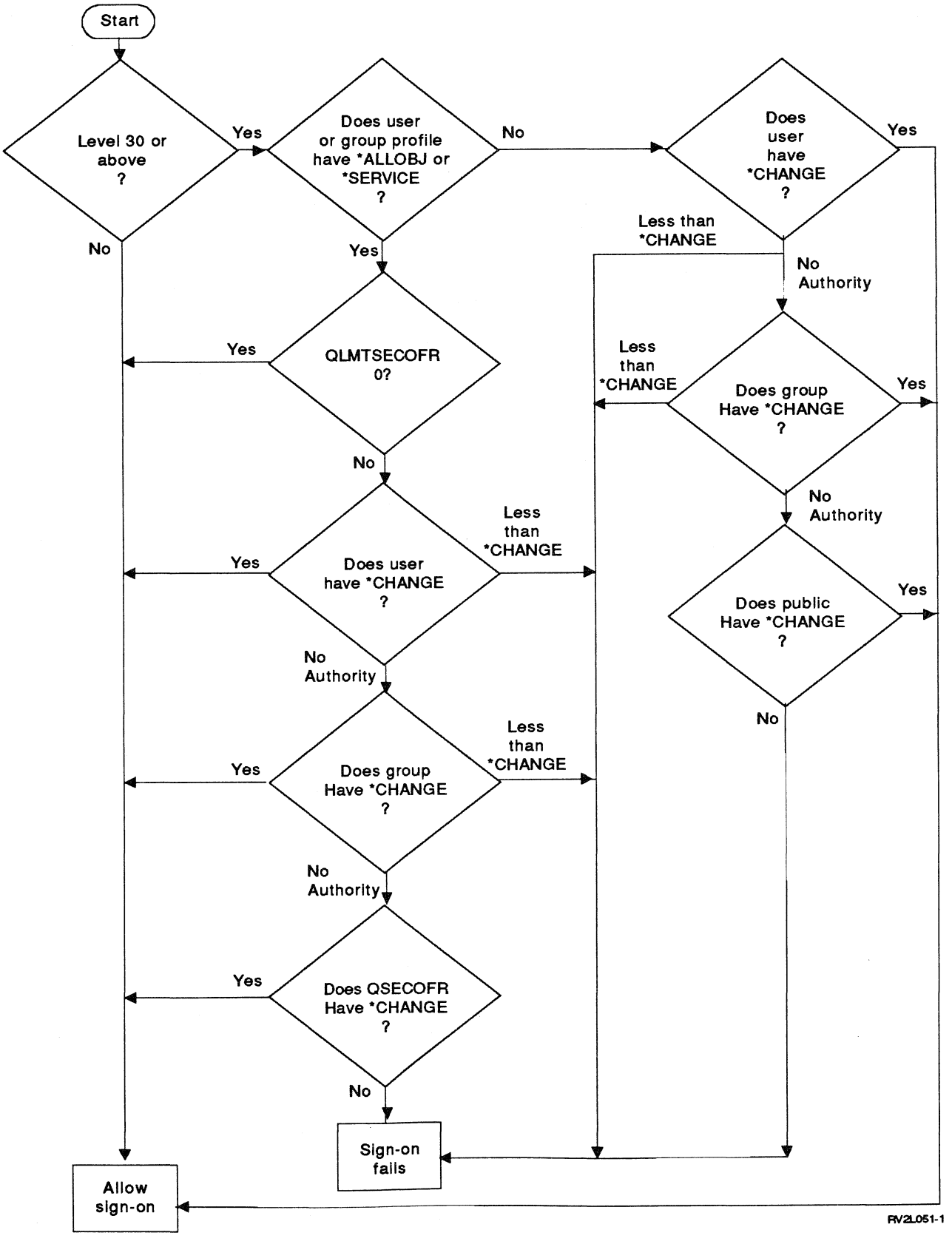


Figure 2-1. Authority Checking for Display Stations

Dedicated Service Tools Considerations

Dedicated service tools (DST) is a group of service functions used to service the system and manage disk devices when the operating system is not running. For more information about recovery operations, see the manual *Backup and Recovery Guide*.

Security Risk

If you activate dedicated service tools (DST) during an initial program load (IPL) of the system, you must end DST using F3 (Exit) on the DST main menu. If you sign off the system after the IPL is complete and do not end DST using F3 (Exit), anyone who has authority to the display station used as the console can press the Sys Req (system request) key and use DST.

The system is shipped with three fixed levels of security for the DST functions. Each level is assigned a password and the menus that are shown are tailored to the functions allowed by the DST function selected at sign-on.

The security levels of DST are shown in the following table.

Table 2-6. Security Levels for Dedicated Service Tools

DST Authority	Default Password	User
Security	QSECOFR	Used by a user to perform all DST functions, including changing the DST passwords.
Full	22222222	Used by a service representative or an experienced system user to provide access to all DST service functions except changing the DST passwords.
Basic	11111111	Used by a service representative or an experienced system operator to provide access to functions that do not access sensitive data. For more information, see the manual *Backup and Recovery Guide*.

DST Password Control

The dedicated service tools are under password security control. To gain access to DST, the keylock switch must be in the Manual position during an attended initial program load (IPL) of the system. The user must enter the DST password. The DST passwords are not the same as the passwords for the operating system or the system service tools (SST) user profiles.

When installing the system, the DST passwords are maintained like the operating system passwords, unless the system storage is first initialized. These passwords can be 1 through 8 characters in length and can contain any letter (A through Z) or number (0 through 9).

The Change Dedicated Service Tools Password (CHGDSTPWD) command allows the security officer (QSECOFR user profile) to reset the DST security password back to its system-shipped default of QSECOFR. Members of the QSECOFR group profile cannot use the CHGDSTPWD command.

Change operating system install security allows you to prevent a user with basic DST authority from installing the operating system. When the operating system install security is in effect, you need one of the following to install the operating system or to use the automatic install from the IPL or Install the System menu:

- Full DST authority password
- Security DST authority password

A user that has security DST authority can use option 4 (Reset system default password) to reset the security officer (QSECOFR) user profile password back to the system-shipped default of QSECOFR.

See the topic "Changing the DST Passwords Using DST" on page 8-4 for the procedure to change DST security passwords.

Service User Profiles Considerations

The system has two IBM-supplied service user profiles: QSRV and QSRVBAS. When using these profiles, consider the following:

QSRV User Profile: This profile is for users who need access to all service tools on the system. It is important to assign someone this responsibility to allow the service representative to sign on and assist a remote service person if a problem occurs that requires service.

QSRVBAS User Profile: This profile is for users who need to use only the basic service tools such as analyze problems, problem log, copy screen image, and print error log.

Security Risk

You should restrict the use of the QSRV and QSRVBAS user profiles because they allow access to sensitive data.

Chapter 3. User Profiles

This chapter provides information about user profiles. It describes the special authorities that can be assigned to a user when you create his user profile and other information used by the system to determine how the user performs certain operations.

All users of the system must have a user profile containing the user's authority to the system. You can tailor the user profile information for each user or you can use the system-supplied defaults.

The following is a list of the information contained in the user profile.

- Name
- Password
- User class
- Assistance level[1]
- Current library[1]
- Initial program[1]
- Initial menu[1]
- Limit capability[1]
- Text[1]
- Special authority
- Special environment[1]
- Display sign-on information
- Password expiration interval
- Set password to expired
- Profile status
- Limit device sessions
- Keyboard buffering[1]
- Maximum storage[1]
- Priority limit[1]
- Job description[1]
- Group profile
  - Group name
  - Owner
  - Group authority
- Accounting code[1]
- Document password
- Message queue[1]
  - Message queue delivery[1]
  - Message queue severity[1]
- Print device[1]
- Output queue[1]
- Attention-key-handling program[1]
- Language identifier (For Version 2 Release 1.1)[1]
- Country identifier (For Version 2 Release 1.1)[1]
- Coded character set identifier (For Version 2 Release 1.1)[1]
- User options[1]
- Public authority
- Owned objects

1 These items are used to tailor a user's environment on the system and are not directly related to security controls.

Authorized objects

Note: Objects owned by a user profile are added to the user profile when a user creates objects or when ownership is transferred to the user. Objects the user has authority to are added to the user profile when the user is given authority to those objects or removed when authority is revoked.

Security Considerations

When creating user profiles, consider the following:

- Several users should not share a single profile because:
  - There is no way of identifying who is using the profile when it is shared by several users.
  - There is no way of tailoring the user profile for a single user when it is shared by several users. For example, each user of the profile would not have an individual message queue assigned to him.

  More than one user can share a user profile to allow the users to learn the system. After they have learned about the system and have been given specific assignments, you should create a user profile for each user.
- Changing passwords is difficult to communicate to all users of a shared profile. To effectively use the capability of requiring a new password after a specified number of days, you should consider assigning unique user profiles to each user.
- Specific authority to objects can be given to each user profile. As the number of specific authorities increases, more time is required to manage those authorities. The time needed to save or restore the system increases when every user profile has many specific authorities. You can keep the number of specific authorities at a minimum by using authorization lists, group profiles and public authority.

The following is an explanation of the information contained in the user profile.

User Profile Name

The **user profile name** identifies the user to the system. User profiles can be assigned the last name and initials of the user (GREENRW), organizational structure (DEPT546MGR), responsibility (PGMR), or group (GROUP777). This name is also known as the user ID that the user types in the *User* prompt on the Sign On display. The user profile name can be a maximum of 10 characters. The characters can be any letter (A through Z), any number (0 through 9), and special characters pound (#), dollar ($), underscore (_), or at (@). The first letter must be alphabetic or special characters pound (#), dollar ($), underscore (_), or at (@).

Note: It is possible to create a user profile so that when a user signs on, the user ID is only numerals. To create a profile like this, specify a Q as the first character, such as Q12345. A user can then sign on by entering 12345 or Q12345 for the *user* prompt on the Sign On display.

For more information about specifying names on the system, see the *CL Programmer's Guide*.

Password

The password is used to verify a user's authority to sign on the system. When password security is active, a password must be specified in addition to a user profile name. Passwords can be a maximum of 10 characters. The rules for specifying passwords are the same as those used for user profile names.

Note: It is possible to create a user profile so that when a user signs on and enters his password, it consists of only numerals. To create a profile like this, specify a Q as the first character, such as Q54321 of the password. A user can then sign on by entering 54321 or Q54321 for the *Password* prompt on the Sign On display.

User profiles whose passwords are specified as *NONE cannot sign on the system at any time.

User Class

User class controls what menu options are shown to the user. This does not necessarily limit the use of commands. The user class can be used to default the special authorities for a user. Each user class has its own set of special authorities given to it.

Special authorities are added or removed based on the user class when changing security levels. See Table 2-1 on page 2-2 and Table 2-2 on page 2-4 for the special authorities by user class for all security levels.

Assistance Level

You can specify which assistance level to use as the default: operational assistant, the system interface, or the expert system interface.

During an active job, you can alter your assistance level by using F21 (Select assistance level) to change your assistance level for a given display. For example, on the WRKACTJOB display, you can use F21 to alter your assistance level from operational assistant to expert system interface.

Current Library

The **current library** is the library that is specified to be the first user library searched for objects requested by the user. You can specify a user's current library where the objects he creates will automatically be placed when *CURLIB is specified as the library to store the object.

If the user is not limited by the LMTCPB parameter, he can specify a different current library:

- At sign-on
- From a menu option
- With the Change Current Library (CHGCURLIB) command
- With the Change Library List (CHGLIBL) command
- With the Change Profile (CHGPRF) command

Initial Program

You can specify the name of a program to call when a user signs on. This program runs before the initial menu, if any, is displayed. You can limit a user to the program specified in his user profile. The limited capability (LMTCPB) parameter determines if the name of the initial program can be specified by the user on the Sign On display. If you limit a user to the program specified, the user cannot specify a different initial program. If a program is not specified, but a menu is specified, that menu is shown.

Note: Parameters cannot be passed to an initial program.

Initial Menu

Security Consideration

You may prefer to give users access to application-defined menus that you create. System-supplied menus may allow users to perform functions on objects that you want to restrict.

You can specify the name of a menu to be shown when the user signs on. If you want to control the user to running only the initial program, you can specify the sign-off (*SIGNOFF) value for the initial menu that signs the user off after the initial program completes.

The limited capability (LMTCPB) parameter determines if the initial menu can be changed by the user. If you limit the user to the menu specified in his user profile, then the user cannot specify another initial menu on the Sign On display.

Limited Capability

You can limit the user's capability to specify the initial program, initial menu, current library, and the Attention-key-handling program. The user is limited to the functions of the menu and/or the program.

You can specify the user's control in several ways.

- Capability is not limited (LMTCPB(*NO)): The initial program, initial menu, and current library value *can* be specified when the user signs on the system. The user *can* specify the initial program, initial menu, current library, or the Attention-key-handling program value in the user's own user profile with the Change Profile (CHGPRF) command, and commands *can* be run from any command line.
- Capability is partially limited (LMTCPB(*PARTIAL)): The initial program and current library value *cannot* be specified when the user signs on the system. The initial menu value *can* be specified and commands *can* be run from any command line. A user *can* specify the initial menu value with the Change Profile command. The user cannot specify the initial program, current library, and the Attention-key-handling program value *cannot* be specified using the Change Profile (CHGPRF) command.
- Capability is limited (LMTCPB(*YES)): The initial program, initial menu, and current library values *cannot* be specified when the user signs on the system. Only a few system-supplied commands *can* be run from a command line. The commands allowed are sign-off (SIGNOFF), send message (SNDMSG), display messages (DSPMSG), display job (DSPJOB), display job log (DSPJOBLOG) and any command created with or changed to have the

ALWLMTUSR(*YES) value. The user *cannot* specify the initial program, initial menu, current library, or the Attention-key-handling program value using the Change Profile (CHGPRF) command.

Security Consideration

To limit a user's access to a command line, specify *YES on the LMTCPB parameter of that person's user profile.

Text

The text in the user profile is used to describe the user profile or what it is used for, such as a department group profile.

Special Authority

Special authority is given to the user when the user profile is created or changed. This authority is used to tell the system the type of actions a user can perform on system resources. If a user is giving another user special authority, he must have the special authority he is giving to the other user. For example, only a user with all object (*ALLOBJ) special authority can give all object special authority to another user.

Consider the following when specifying special authority:

- **All object** (*ALLOBJ) special authority allows a user access to all system resources even if he has no authority given to him for the resource.

  Security Risk

  Before you give a user all object special authority, you should consider if the user really needs this authority. Giving a user this authority is a **security risk** because it allows the user to use or delete any object. The user is not limited to authority for specific objects.

- **Save system** (*SAVSYS) special authority allows a user to do save and restore operations for all resources on the system[2].

  Security Consideration

  Before you give a user save system special authority, be aware that they also can free storage when saving objects. Freeing storage deletes the data portion of the objects.

- **Job control**[3] (JOBCTL) special authority allows a user to:
  - Change, cancel, hold, and release all files on output queues[4]
  - Hold, release, and clear job queues and output queues[4]
  - Hold, release, change, and cancel other users' jobs
  - Start writers[4]
  - Change the running attributes of a job, such as the printer for a job

---

2 The user must also have authority to the devices.

3 A user must have job control special authority to change his or her run priority (RUNPTY parameter) on the Change Job (CHGJOB) command. The user can change the job priority (JOBPTY) and the output priority (OUTPTY) parameter without job control special authority.

4 The output queue must also be specified as OPRCTL(*YES).

- Stop subsystems
- Perform an initial program load (IPL)

- **Security administrator** (*SECADM) special authority allows a user to:
  - Add users to the system distribution directory (this includes the right to create and change user profiles for OfficeVision/400 users)

    You must have security administrator authority to use the Create User Profile (CRTUSRPRF), Change User Profile (CHGUSRPRF), and Delete User Profile (DLTUSRPRF) commands. This authority is not required for the Change Profile (CHGPRF) command.
  - Display authority for documents or folders
  - Add and remove access codes to the system
  - Give and remove a user's access code authority
  - Give and remove permission for users to work on another user's behalf
  - Delete documents and folders
  - Delete document lists
  - Change distribution lists created by other users
  - Change security-related system values and network attributes

  Only a user with *SECADM and *ALLOBJ special authority can give another user *SECADM special authority.
- **Spool control** (*SPLCTL) special authority allows a user to control spool functions, such as cancel, delete, display, hold, and release other users' spooled files. If you have a user who needs access to any spooled files, specify *SPLCTL for that user.
- **Service** (*SERVICE) special authority allows a user to perform the display and alter service functions. The dump function can be performed without this authority.

Security Risk

Before you give a user service special authority, you should consider if the user really needs this authority. Giving a user this authority is a **security risk** because it can allow the user to access sensitive data.

Display Sign-On Information

The display sign-on information parameter specifies if the information from the last sign-on is shown. This display allows users to see the information, such as the date of the last sign-on and the sign-on attempts that were not valid. If the password is due to expire in seven days or less, the number of days until the password expires is shown.

Password Expiration Interval

Requiring users to change their passwords after a specified length of time reduces the risk of an unauthorized person accessing the system. The password expiration interval parameter controls the number of days that a valid password can be used before it must be changed. If the password is not changed in the number of days specified, the user cannot sign on until the password is changed. The system warns the user that the password is about to expire when the user signs on the system.

Set Password to Expired

The set password to expired parameter allows the security officer to indicate in the user profile that the user's password is expired and must be changed the next time the user signs on. After the user changes his password, this value is reset to *NO.

This parameter can be used when a user cannot remember his password and the security officer must give him a new one. Requiring the user to change the password assigned by the security officer during sign-on prevents the security officer or the security administrator from knowing the new password. This prevents the security officer or security administrator from being able to sign on the user's display station using that user's password.

Status

The status parameter indicates if the profile is enabled or disabled. If the profile status is enabled, the profile is valid for sign-on. If the profile status is disabled, an authorized user has to enable the profile again to make it valid for sign-on.

Limit Device Sessions

The limit device sessions parameter specifies whether or not a user is limited to one device session. This value does not restrict the use of the System Request menu nor a second sign-on from the same device.

Keyboard Buffering

This parameter specifies the keyboard buffering value used when a job is initialized for this user profile. The new value takes effect the next time the user signs on. The keyboard buffer value can also be set by a user application using the QWSSETWS program.

Special Environment

Special environment determines the environment the user operates in after signing on. The user can operate in the AS/400, the System/36, or the System/38 environment. If the user's routing program is the AS/400 command processor (QCMD), the system uses this value to determine which environment to use. Batch jobs also use the special environment value if the user's routing program is the AS/400 command processor (QCMD). If a user's routing program is the System/38 command processor (QCL), then the System/38 environment is used.

If you have users who are running System/36 applications and programs most of the time, specify the System/36 environment. If the user runs in the System/36 environment only part of the time, specify the AS/400 environment; the user can specify the STRS36 command to start the System/36 environment when he needs it.

The following shows how the system determines the environment:

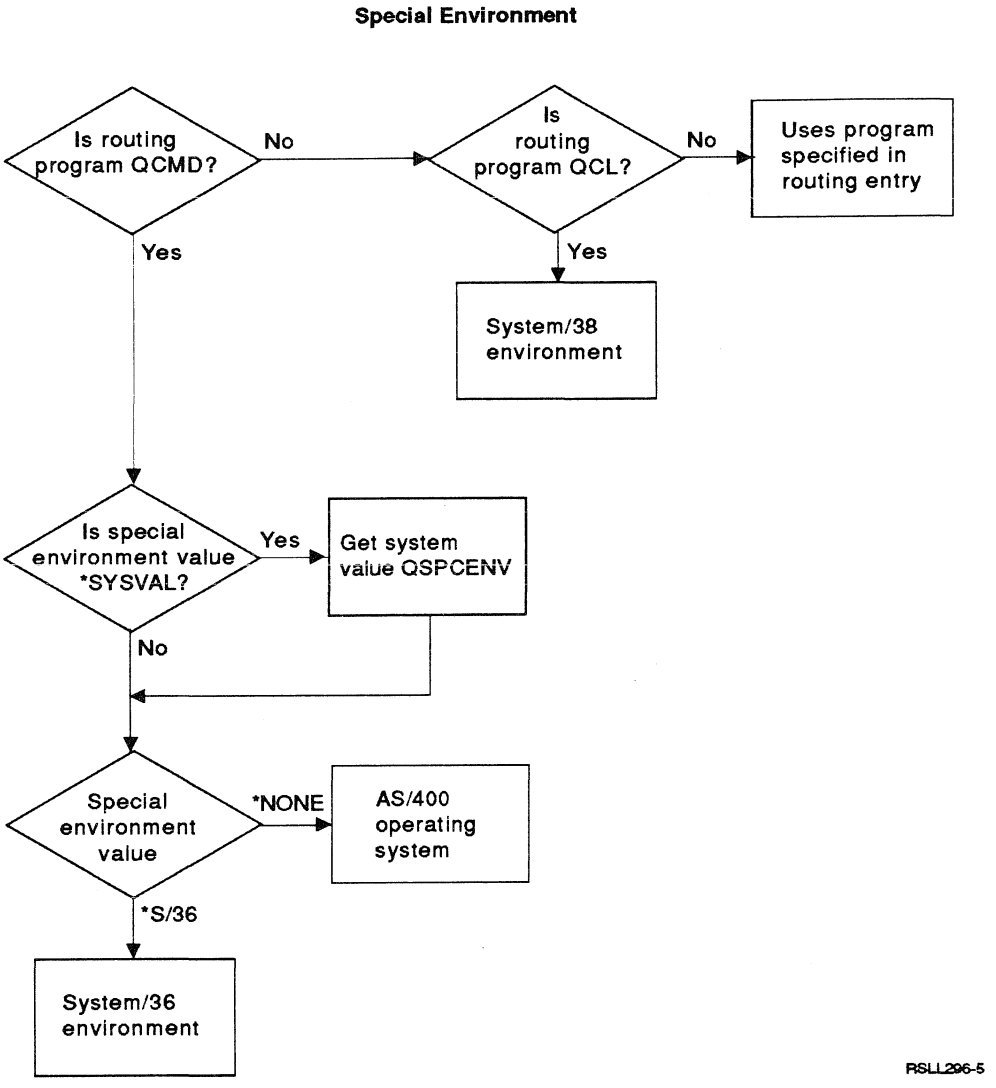


Figure 3-1. Determining the Special Environment

Maximum Storage

You can specify the maximum amount of auxiliary storage that is used to store permanent objects that are owned by a user profile, including objects placed in the temporary library during a job. If the storage needed is greater than the maximum amount specified when the user attempts to create an object, the object will not be created.

The maximum storage specified is in kilobytes. When planning maximum storage for user profiles, you should consider the following system actions:

- A restore operation first assigns the storage to the user doing the restore operation, and then transfers the objects to the owner. If you have users who do large restore operations, you should specify MAXSTG(*NOMAX) for those users.
- The user profile that creates a journal receiver is assigned the storage as the receiver size grows. If new receivers are created, the storage continues to be assigned to the user profile that owns the active journal receiver. If

you have users who own active journal receivers, you should specify MAXSTG(*NOMAX) for those users.

- Users that transfer all created objects to their group profile must have adequate storage in the user profiles to contain any created object before the object is transferred to the group profile.
- The owner of a library is assigned the storage for the descriptions of the objects that are placed in a library, even when the objects are owned by another user profile. Examples of such descriptions are text and program references.
- Storage is assigned to the user profile for temporary objects that are used during the processing of a job. Examples of such objects are commitment control blocks, file editing spaces, and documents.

Priority Limit

You can specify the maximum scheduling priority limit that the user is allowed to have for each job that he submits to the system, and you can limit the job scheduling priority and output priority that any job running under his user profile can have. Values specified for job priority and output priority of any job command cannot be more than the priority limit specified for the user profile under which the job is run.

If you have a user who submits jobs that use many system resources, you can specify the scheduling priority of 9 (lowest priority) for that user. Jobs that use many resources should have the least priority so that other users' jobs can run.

If you have a user who needs to move jobs to the front of the job queue, you can specify the scheduling priority of 1 for that user.

Job Description

You can specify the name of the job description that contains a specific set of job-related attributes, such as batch queue, scheduling priority, routing data, and message queue severity, which can be used by a user when he submits a batch job. The attributes determine how each job is run on the system. The same job description can be used by many jobs.

Group Profile

Group Name: You can specify the name of a group profile whose authorities may be used for a job. The group's authority for the object is used only if the group member has no specific authority for the object.

To specify a group profile on the Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command, you must have object management (*OBJMGT) and change (*CHANGE) authorities to the group profile specified. However, *OBJMGT authority from a program that adopts the owning user profile's authority is not allowed.

For more information about the group profile function, see the topic "Group Profiles" on page 4-9.

Owner: You can specify the owner of objects created by this user. You can specify this user profile as the owner or specify the group profile as the owner. Owner is used only if a group profile is specified.

For more information about object ownership, see the topic "Object Ownership" on page 4-3.

Group Authority: You can specify the authority given to the group profile for newly created objects. Group authority is used only if a group profile is specified. If a group profile is specified as the owner, group authority must be *NONE.

Accounting Code

You can specify an accounting code used by jobs that get their accounting codes from the user profile instead of the system. Accounting codes are associated with job and printed accounting data for the system resources used for that job when accounting is active.

Document Password

You can specify a document password for the user to protect the distribution of personal mail from being viewed by people working on his behalf when using the Document Interchange Architecture (DIA).

Message Queue

You can specify the name of a message queue for a user. A **message queue** is a list on which messages are placed when they are sent to a person or a program. A message queue is used when a user sends or receives messages. If a message queue does not exist when a user signs on, it is automatically created. If the name of a message queue is changed in a user profile using the Change User Profile (CHGUSRPRF) command, then that message queue is also created. If you specify a different message queue in the user profile, the previous message queue is not deleted. If a user profile is created with a password of *NONE, a message queue is not created.

For more information about message queues, see the *Operator's Guide*.

Delivery: You can specify one of four ways that messages sent to the message queue are delivered for the user profile.

- Messages can be **held** (*HOLD) in the message queue until they are requested by the user or program.
- The job to which the message queue is assigned is **interrupted** (*BREAK) when a message is received at the message queue. If the job is an interactive job, the audible alarm is also sounded (if one is installed).
- The job to which the message queue is assigned is **notified** (*NOTIFY) when a message is received at the message queue. For interactive jobs at a display station, the audible alarm is sounded and the message-waiting light is turned on.
- Messages requiring replies are answered with a **default** (*DFT) reply, and information-only messages are ignored.

Severity: You can specify the lowest severity code that a message can have and still be delivered to a user in break or notify type of delivery. Messages received at the message queue whose severities are lower than that specified do not interrupt the job or turn on the message-waiting light; they are held in the queue until they are displayed by the Display Message (DSPMSG) command.

Printer Device

You can specify the printer used to print the output for this user. The spooled file is placed on an output queue with the same name as the printer when the output queue (OUTQ) is specified as the printer device (*DEV).

Output Queue

An **output queue** contains a list of spooled files that are to be written to an output device.

You can specify the output queue for the user. An entry is placed on the output queue for each spooled output file. The output queue must already exist.

Attention-Key-Handling Program

An **attention-key-handling program** is a user-defined program that is called when the user presses the Attention (Attn) key during an interactive job. You can specify the program that is used as the Attention-key-handling program for the user. The program is active only when the user's routing program is the system-supplied command processor, QCMD. The Attention-key-handling program is set on before the initial program (if any) is called and is active for both the program and the menu. If the initial program changes the Attention-key-handling program, the new Attention-key-handling program remains active only until the initial program ends. When control returns and the command processor (QCMD) calls the menu, the original Attention-key-handling program becomes active again. If the Set Attention-Key-Handling Program (SETATNPGM) command is run from a command line or an application, the new Attention-key-handling program specified overrides the original Attention-key-handling program.

The limit capability (LMTCPB) parameter determines if a different Attention-key-handling program can be specified by the user with the Change Profile (CHGPRF) command.

Language Identifier (For Version 2 Release 1.1)

You can specify the language identifier to be used by the system for the user.

Country Identifier (For Version 2 Release 1.1)

You can specify the country identifier to be used by the system for the user.

Coded Character Set Identifier (For Version 2 Release 1.1)

You can specify the coded character set identifier to be used by the system for the user.

User Options

The user options parameter allows you to specify what the system shows the user. You can specify that the system:

- Show displays for the experienced or inexperienced users. The experienced user will see all fields on the display.
- Reverse the way the Page Up and Page Down keys normally work.
- Send a message to your message queue when a spooled file you own is printed.
- Show or not show the user status messages.

- Show command parameters when prompted on a command display.
- Show full screen help or in windows.

For example, if a user needs to see the parameter names of a command, you can specify USROPT (*CLKWD) to show the parameters to a user automatically.

Authority

You can specify the authority other users can have to the user profile. This authority is called public authority. Public authority is given to users who do not have any specific (private) authority to the user profile and whose group profile has no specific authority to the user profile.

The values for authority can be *ALL, *CHANGE, *USE, or *EXCLUDE. Each authority (except *EXCLUDE) is a combination of one or more of the object authorities and data authorities.

Add authority is required for a user profile to transfer ownership of objects to that profile. Read authority is required to name the user profile in a job description. Use authority is required to the user profile to submit a job when the job description specifies a user profile name.

For more information about object authorities and data authorities, see "Specific Authority" on page 4-1.

Chapter 4. Resource Security

This chapter provides information about resource security. **Resource security** consists of the security measures you use to authorize users to specific objects. Authority can be specified in two ways: privately or publicly. **Private** authority is the authority given specifically to a user for a resource. **Public** authority is the authority used when the following occur:

- A user has no authority given specifically for an object
- A user is not on the authorization list (if one is specified for an object)
- A user's group profile (if one is specified in the user profile) does not have any authority given specifically for an object
- A user's group profile is not on an authorization list that has authority for an object

Resource security consists of object authority and data authority. These authorities are specified when giving a user authority for an object using the Edit Object Authority (EDTOBJAUT) or the Grant Object Authority (GRTOBJAUT) command.

Note: When an object is deleted from the system, all private and public authority for the object is also deleted from the system. When the object is created again, the authority must be granted to each user who needs authority for the object.

Specific Authority

Specific authority determines how a user or the public can use an object. Specific authority can be defined two ways: the user can define authority by combining one or more object authorities with one or more data authorities, or he can specify one of the authorities defined by the system that is a combination of object authorities and data authorities. The following topics describe the object and data authorities you can specify.

Authority Defined by the User

Authority defined by the user is shown on the Display Object Authority display as USER DEF. A maximum of eight objects and data authorities can be specified.

Object Authority

The following is a description of the object authorities that can be specified.

Object operational (*OBJOPR) authority allows the user to look at the description of an object and use the object as determined by the data authorities that the user has to the object.

Object management (*OBJMGT) authority allows the user to specify the security for the object, move or rename the object, and add members to database files.

Object existence (*OBJEXIST) authority allows the user to control the object's existence and ownership. This authority is necessary for users who want to delete the object, free storage of the object, perform save and restore operations for the object, or transfer ownership of an object. (If a user has save system (*SAVSYS) special authority, he does not need object existence authority to perform save and restore operations.)

Authorization list management (*AUTLMGT) authority allows the user to add and remove users and their authorities on an authorization list. A user with authorization list management authority can remove a user profile name from the list only if he has the same authorities as the user profile name being removed. He can add, change, or remove authority only if he has the same authorities being added, changed, or removed. For example, a user with *CHANGE authority and *AUTLMGT authority can remove users with *CHANGE authority or *USE authority because he has object operational authority and all the data authorities. He cannot remove a user with *ALL authority because he does not have object existence authority or object management authority. He can add a user and give him *CHANGE authority or *USE authority, but he cannot add a user and give him *ALL authority.

Data Authority

The following is a description of the data authorities that can be specified:

Read (*READ) authority allows the user to display the contents of an object or run a program.

Add (*ADD) authority allows the user to add entries to an object. (For example, adding job entries to a job queue or adding records to a file.)

Update (*UPD) authority allows the user to change the entries in an object.

Delete (*DLT) authority allows the user to remove entries from an object. (For example, removing messages from a message queue or records from a file.)

Subset of Authorities Defined by the System

The system has defined four authorities that allow the user to select a subset of object authorities and data authorities for an object. This subset combines one or more object authorities with one or more data authorities (see Table 4-1 on page 4-3). Authorization list management authority can be specified with one of the system-defined authorities only for an authorization list object. The only exception is exclude authority. If *EXCLUDE authority is specified, no other authority can be specified.

The following is a description of the system-defined authorities that can be specified:

All (*ALL) authority provides all the object authorities and data authorities. The user can control the object's existence, specify the security for the object, change the object, and perform basic operations on the object such as run a program or display the object's description and contents.

Change (*CHANGE) authority provides object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

Use (*USE) authority provides object operational authority and read authority. The user can run a program or display the object's description or contents. The user is prevented from changing the object.

Exclude (*EXCLUDE) authority prevents the user from accessing the object. If this authority is specified, no other authority can be specified.

The following table shows the subsets of object authorities and data authorities:

Table 4-1. System-Defined Authority

Authority	Object OPR	MGT	EXIST	Data READ	ADD	UPD	DLT
*All	X	X	X	X	X	X	X
*Change	X			X	X	X	X
*Use	X			X			
*Exclude	No	authority					

Object Ownership

Each object is assigned an owner when it is created. The owner is either the user who creates the object or the group profile if the member user profile has specified that the group profile should be the owner of the object. When the object is created, the owner is given all the object and data authorities to the object. When ownership of an object is given to another user profile, the original owner has the option to keep all of the object and data authorities he had before the change of ownership, or remove all the authority he had for the object.

The owner of an object always has all the authority for the object unless any or all authority is removed specifically. As the owner, he has the authority to give any authority to any user for his objects. He can also give himself any authority that was previously removed. The owner may, for example, remove some of his specific authority as a precautionary measure, and then, when he needs that authority, he can again give the same authority to himself. For example, if a file exists that contains critical information, the owner can remove his object existence authority to prevent him from accidentally deleting the file.

To transfer ownership, any user (including the object's present owner) must have all of the following:

- Object existence authority for the object (except for an authorization list)
- All object authority or ownership if the object is an authorization list
- Add authority for the new owner's user profile
- Delete authority for the present owner's user profile

Security Consideration

A user with all object (*ALLOBJ) special authority has complete authority for all objects; therefore, he can transfer the ownership of any object.

Before a user profile is deleted, the objects owned by that user must be given to new owners. Otherwise, the objects must be deleted because a user profile cannot be deleted while it still owns objects.

Objects on the system sometimes become damaged and must be consequently deleted. If a user profile becomes damaged and is deleted, its objects will not have an owner. This is corrected with the Reclaim Storage (RCLSTG) command; the objects are given to the default owner (QDFTOWN) user profile. In all other instances, an object cannot exist on the system without an owner.

See the topic "Default Owner (QDFTOWN) User Profile" on page 4-6 for more information about the QDFTOWN user profile.

When changing an object's owner, you have the option to remove the former owner's authority.

In the following example, the present owner is changing the owner of menu program ORDMNU for the order department from HANDERSON to BWALTON. The Change Object Owner (CHGOBJOWN) command is entered as follows:

```
CHGOBJOWN       OBJ(DSTPRODLB/ORDMNU)  OBJTYPE(*PGM)
                NEWOWN(BWALTON) CUROWNAUT(*REVOKE)
```

If you display the object's authority, it appears as follows if you used F11 (Display Detail).

Object Authority Before Ownership Change

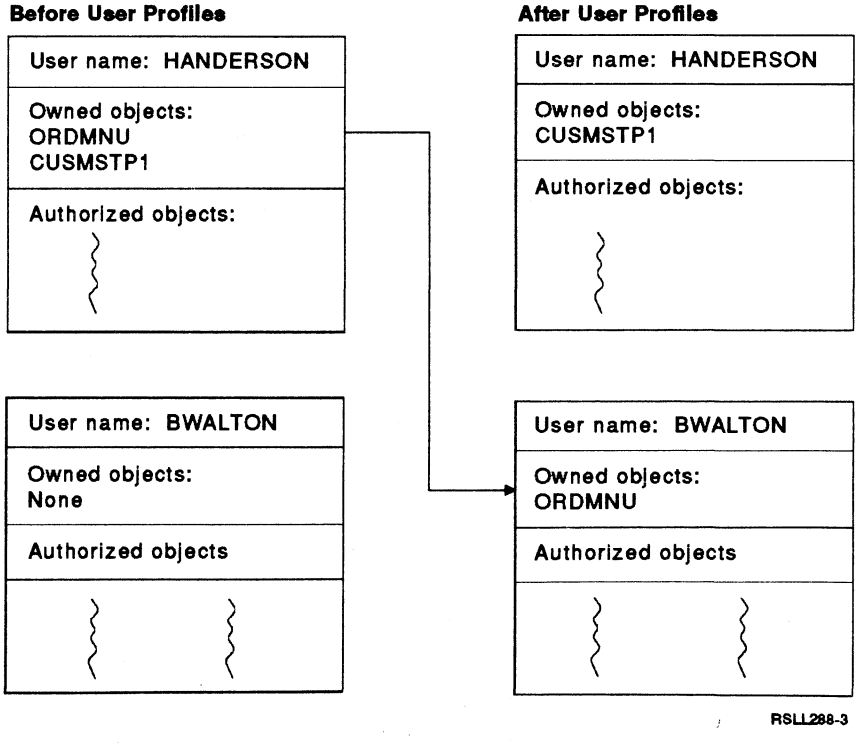
```
                        Display Object Authority

Object . . . . . .: ORDMNU          Object Type . . . .: *PGM
  Library. . . . . .: DSTPRODLB     Owner . . . . . .: HANDERSON

Object secured by authorization list . . . . . : *NONE

               Object     -----Object------   ---------Data---------
User           Authority   Opr  Mgt   Exist   Read  Add  Update  Delete
HANDERSON      *ALL         X    X      X      X     X     X       X
*PUBLIC        *USE         X                  X

Press Enter to continue.                                          Bottom

F3=Exit  F11=Nondisplay detail   F12=Cancel  F17=Top  F18=Bottom

(C) COPYRIGHT IBM CORP. 1980, 1991.
```

If you display the object authority again, it changes to:

Object Authority After Ownership Change

```
                        Display Object Authority

Object . . . . . .: ORDMNU          Object Type . . . .: *PGM
  Library. . . . . .: DSTPRODLB     Owner . . . . . .: BWALTON

Object secured by authorization list . . . . . : *NONE

               Object     ------Object------  --------Data-----------
User           Authority  Opr  Mgt   Exist   Read  Add  Update  Delete
BWALTON        *ALL        X    X      X       X     X    X       X
*PUBLIC        *USE        X                   X

Press Enter to continue.                                        Bottom

F3=Exit  F11=Nondisplay detail  F12=Cancel  F17=Top   F18=Bottom

(C) COPYRIGHT IBM CORP. 1980, 1991.
```

The user profiles change like this:



Notice that the old owner HANDERSON no longer has authority to the menu program ORDMNU because the old owner's authority was revoked (*REVOKE is the default on the Change Object Owner (CHGOBJOWN) command).

Notes:

1. If you want to change ownership of a device description for a display station, the display station associated with that device description must be varied on and you should run the Change Object Owner (CHGOBJOWN) command from the display station. You can run the CHGOBJOWN command from a different display station by allocating the device description of the display station whose ownership is being changed using the Allocate Object (ALCOBJ) command and then deallocate it using the Deallocate Object (DLCOBJ) command when the change request is completed. However, you cannot allocate the device description if it is in use.
2. You cannot use the Change Object Owner (CHGOBJOWN) command for filed documents and folders (object type *DOC and *FLR). For information about the command used to change ownership of these object types, see Appendix D, "Authority Required for Objects Used by Commands."

Default Owner (QDFTOWN) User Profile

The Default Owner (QDFTOWN) user profile is an IBM-supplied user profile that is used when an object has no owner or when a program that adopts the owner's authority is restored by a user who is not the owner or a user who does not have all object (*ALLOBJ) and security administrator (*SECADM) special authorities. Objects with no owner can result from:

- Damage to a user profile that was subsequently deleted.
- Exceeding the maximum storage limit for the user profile that owns an authority holder that has the same name as a file being moved, renamed, or whose library is being renamed.
- Restoring a program that adopts the owner's authority but the owner no longer exists.
- When someone other than the owner or the security officer restores a program that adopts the owner's authority.
- Restoring a program that needs to be translated but the translation fails (template does not exist, restricted instructions used). The attempt to retranslate a program does not happen in all cases.
- Restoring an object when the owner does not exist.

The system supplies the QDFTOWN user profile because all objects must have an owner. Only a user with *ALLOBJ special authority can display and access this user profile and transfer ownership of objects associated with the QDFTOWN user profile.

Grouping Users

You may have a situation where you want to allow several users to access the same objects. This topic discusses grouping users and resources for authority using authorization lists and group profiles. Grouping users and resources simplifies management of authority.

Authorization Lists

An authorization list contains a list of users and the authorities that each user has to all the objects the list secures. Then, when you specify the authority for an object or a set of objects, you can specify an authorization list name. This way, the same authorization list can be used for many objects without giving each user authority to the object individually. Each user on the authorization list can have a different authority to the set of objects the list secures.

The advantages of using an authorization list follow:

- Authorization lists allow all users on the list to be given authority for an object in one operation.
- Adding, changing, and removing users on an authorization list apply to all objects secured by the authorization list.
- Authorization lists reduce the number of authorities on the system.
- Authorization lists provide a way to remember authorities when an object is saved. When an object is saved that is secured by an authorization list, the name of the authorization list is saved with the object. If the object is deleted and restored to the **same** system, it is automatically linked to the authorization list again. If the object is restored on a different system, the authorization list is not linked.

Notes:

1. Authorization lists cannot be used to specify the security for a user profile or another authorization list.
2. Only one authorization list can be specified for an object.
3. Only the owner of the object, a user with all object (*ALLOBJ) special authority or a user with all (*ALL) authority, can add or remove the authorization list for an object.
4. Objects in the system library (QSYS) can be secured with an authorization list. However, installing a new release of the system removes the authority granted by an authorization list from those objects in QSYS. After installing the system, the objects in library QSYS must be secured again by the authorization list.

If you specify an authorization list (AUT(authorization-list-name)) on a create command when the object is created, the public authority value of the authorization list that you specified on the AUT parameter is used because the public authority is set to *AUTL for the object.

A user's authority specified on the authorization list can be overridden by specifically giving the user authority for the object. Any private authority given to the user overrides the authority specified in the authorization list for the user. For example, user TOM is on an authorization list with *USE authority but also has *READ authority given to him for the object; TOM will only have *READ authority. For more information about authority checking, see "Authority Checking" on page 4-29.

The following figure shows an illustration of an authorization list.

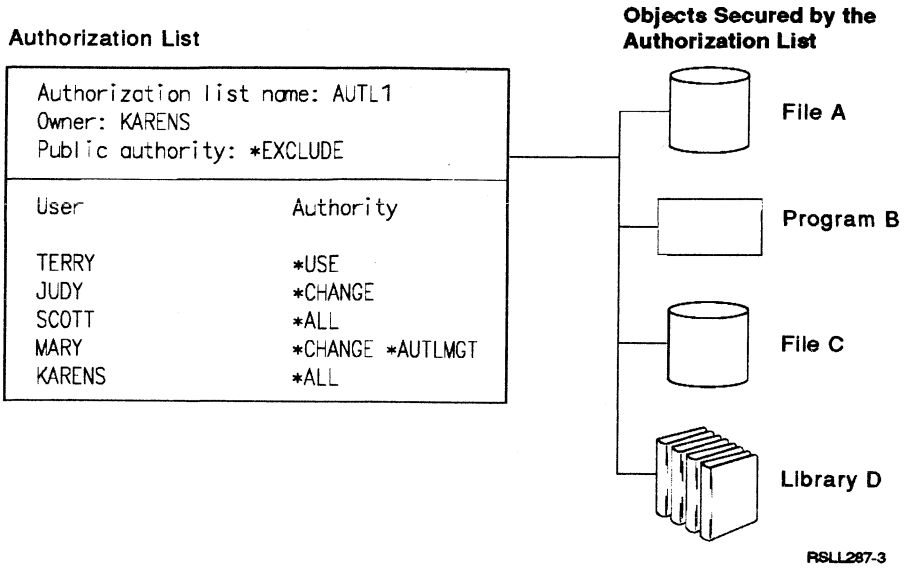


Figure 4-1. Example of an Authorization List

In the example, KARENS has created an authorization list and has automatically been given all (*ALL) authority to those objects. By specifying the authorization list for the four objects she has created as AUTL1, then:

- User TERRY has *USE authority for all four objects. Terry can display the contents of the files or run Program B. TERRY cannot change or delete any objects.
- User JUDY has *CHANGE authority for all four objects. Judy can look at the description of the object and add, delete, and update entries in the object. Judy cannot delete any of the four objects.
- User SCOTT has *ALL authority for all four objects. Scott can control the objects' existence, specify the authority for the objects, and change the objects.
- User MARY has *CHANGE authority and also *AUTLMGT authority. Mary can do the same operations as Judy. In addition, she can add or remove users who have *CHANGE authority or less from the authorization list, rename the authorization list, and change a user's authority on the authorization list. Mary cannot add users or change users' authorities nor give or remove an authority that she does not have.
- All other users who do not have any authority cannot use the four objects because the public authority is *EXCLUDE.

If a new user, BETTY, is added to the authorization list (ADDAUTLE command) with *CHANGE authority, she is given authority for all four objects. If a new object (File E) is secured by the authorization list, GRTOBJAUT AUTL(AUTL1), all users on the list have the same authority to File E as they have to other objects secured by this list.

If you display the authorization list, DSPAUTL AUTL(AUTL1), it looks like this:

```
                    Display Authorization List

Authorization list .: AUTL1                  Owner . . . . . .: KARENS
   Library . . . . .: QSYS

                Object     AUTL
User          Authority    Mgt
TERRY         *USE
BETTY         *CHANGE
JUDY          *CHANGE
SCOTT         *ALL
MARY          *CHANGE       X
KARENS        *ALL
*PUBLIC       *EXCLUDE



Press Enter to continue.                                          Bottom


F3=Exit  F11=Display detail  F12=Cancel
F15=Display auth list objects  F17=Top   F18=Bottom
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

To display the detailed authorities press F11 (Display Detail). To display the list of objects secured by AUTL1, press F15 (Display auth list objects). Users can display an authorization list as long as their private authority or the public authority is not *EXCLUDE.

When an object is saved and then deleted, consider the following:

- When the object is restored to the *same* system, the object is linked to the authorization list.
- When the object secured by the authorization list is restored to a *different* system, it is linked to the authorization list when ALWOBJDIF(*ALL) is specified on the Restore command.

For more information about the authorization list values, see the topic "Planning an Authorization List" on page 7-41.

Group Profiles

Group profiles provide a way to simplify authority management. Group profiles make it easier to change authorities that affect every member of the group. This approach also makes it easier to add a user to, delete a user from, or move a user to a new department. If the user moves to a new department, the authority change requires only changing the user profile GRPPRF parameter to specify a new group profile name.

You can create a user profile specifically as a group profile or you can specify an existing user profile as a group profile, including some (but not all) of the IBM-supplied user profiles, such as QPGMR. When a group profile is specified on the Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command, each member is automatically granted change (*CHANGE) authority and object management (*OBJMGT) authority to the group profile. To set up a group profile, you change the GRPPRF parameter in the user profile to reference a group profile name. The name in the GRPPRF parameter makes the specified user profile a group profile. The profile must exist before it can be specified as a group profile.

A group profile cannot be a member of another group profile. A user can be a member of only one group profile. However, a group can have authority to multiple objects. In this way, the number of authorities on the system is reduced.

A member of the group can share all the authorities given specifically to the group, as well as the special authorities of the group. For example, if a group has authority to a set of objects and has save system (*SAVSYS) special authority, (see Figure 4-2), the members of the group get the authority specified for the group for the list of objects along with any special authorities given to the group.

Any member of a group requiring authority different from the group authority for the objects can be specifically given authority using the Grant Object Authority (GRTOBJAUT command) or the Edit Object Authority (EDTOBJAUT) command.

In Figure 4-2 the authorities given specifically to a group are passed to the members unless a member of the group was specifically given his own authority for the same object.

- Members of the group share special and object authorities.

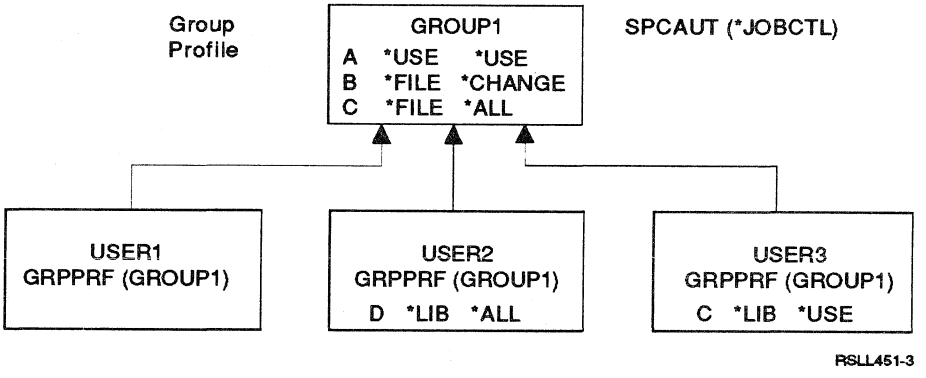


Figure 4-2. Objects Secured by a Group

- Private authorities add to or override the group's authorities to objects. See Table 4-2.

Table 4-2. Special Authority and Private Authority

Authority	USER1	USER2	USER3
GROUP1	*JOBCTL	*JOBCTL	*JOBCTL
GROUP1	A B C	A B C	A B
Private		D	C

  - USER2 has *ALL to D (in addition to the group)
  - USER3 has *USE to C (overrides the group)

When an object is saved, the group profile is not affected. If an object is deleted and then restored, authority for the object must be granted to the group profile again if the group is not the owner of the object.

When a member of a group starts a job, the parameter values for group profile (GRPPRF), owner (OWNER), and group authority (GRPAUT) are copied from the member's user profile at the start of the job and are stored as part of the job information. If these values are changed while the job is running, the change does not affect the current job.

Security Consideration

When transferring a member from one group profile to another, it is possible for the transferred member of the group to have authority for objects he created while a member of the former group profile that he should no longer have. Consider the following when transferring a member from one group to another:

- If user profile TOM specified OWNER(*USRPRF) while a member of the former group profile GRPOLD, TOM remains the owner of the objects he created even if he is now a member of a new group GRPNEW.
- If user profile TOM specified anything other than GRPAUT(*NONE) for objects that TOM created while a member of the former group profile, GRPOLD will still keep authority for those objects.

If user profile TOM no longer needs authority to the objects he created when he was a member of GRPOLD group profile, ownership of those objects should be transferred to the GRPOLD group profile.

If the GRPOLD group profile no longer needs authority for the objects TOM created, the group's authority should be removed from the objects using the Revoke Object Authority (RVKOBJAUT) command.

Group Ownership of Objects

In general, when creating a department group or using an existing user profile such as QPGMR as a group, you should decide who the owner of objects created by the group members should be. If the owner of the objects created by a member of the group should be the member, the OWNER parameter must use the default value *USRPRF, and the GRPAUT parameter must specify what authorities (if any) are given to the group profile. OWNER and GRPAUT are specified in the member's user profile and not in the group profile. If a member creates an object and GRPAUT(*ALL) is specified in the member's user profile, then the system grants *ALL authority to the group. This authority becomes a private authority for the group.

For example, an object can be created with the default public authority of AUT(*USE), and the member user profile that creates the object can specify GRPAUT(*ALL). Users with no other authority to the object are given the public authority of *USE and all the members of the group are given *ALL authority. An alternative is to create the object with the public authority of AUT(*EXCLUDE) and give change authority to the group by specifying GRPAUT(*CHANGE). This means that users outside the group that have no other source of authority for this object cannot use it. Members of the group can read, add, change, and delete information in the object.

This alternative also allows you to display which user created the object and allows any member of the group access to the object. The public authority is normally given a value that prevents other users from changing or deleting an object.

Security Consideration

Specifying OWNER(*USRPRF) and GRPAUT(*ALL) gives *ALL authority specifically to the group profile. Changing GRPAUT(*ALL) to GRPAUT(*NONE) or GRPAUT(*EXCLUDE) does not remove the authority previously given for the objects.

Specifying GRPAUT(*NONE) allows a member to use the public authority specified for the object. Specifying GRPAUT(*EXCLUDE) does not allow a member to use the public authority specified for the object.

If a member's authority for a newly created object must be different than other members of the group, the member's user profile that is creating the object should specify OWNER(*USRPRF) GRPAUT(*EXCLUDE). The authority for other members of the group can be controlled by the Grant Object Authority (GRTOBJAUT) and Revoke Object Authority (RVKOBJAUT) commands.

If OWNER(*GRPPRF) is specified, then all objects created by the group members are owned by the group profile and any member of the group has the authority to access the object. This gives consistency in ownership, but does not allow you to display which user created the object.

Group Profile Methods

You can use either of two methods to make a user profile a group profile. One method is to create the group profile (using the CRTUSRPRF command) specifically for a group of users. Another method is to use an existing user profile as a group.

Method 1: You can create a group profile for a specific department such as DEPT547 if the department is made up of several users with common authorities.

Security Consideration

If you plan to use group profiles, use a naming convention for the group profile, such as GROUPXXX or DEPTXXX that allows you to identify the profile as a group. Then, when you are displaying the list of authorized users for an object, it is easier to identify that there are group profiles with additional users. The naming convention should be used for all group profiles.

In the following example, the user profile DEPT547 becomes a group profile when the user profiles that are members of the group are changed to specify GRPPRF(DEPT547). The system knows that DEPT547 is now a group profile and remains so as long as at least one user profile is specified as GRPPRF(DEPT547). A group profile cannot be deleted if it has one or more members associated with it.

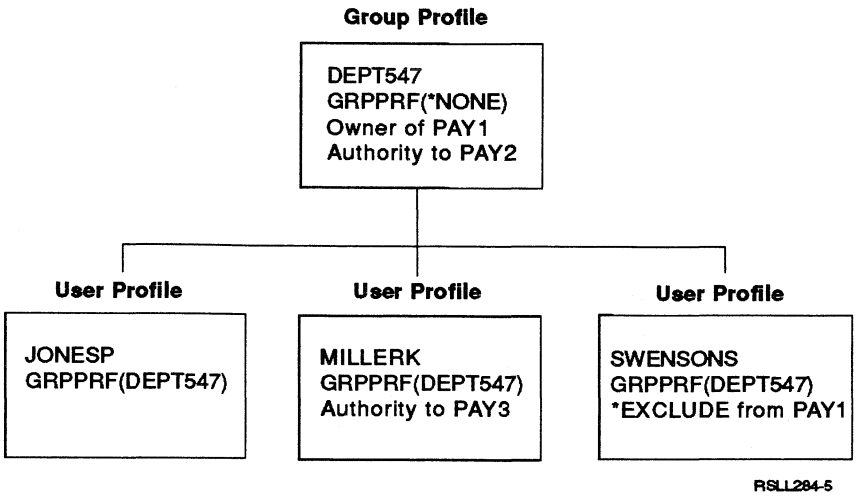


Figure 4-3. Group Profile Example

When JONESP, MILLERK, and SWENSONS start a job, they use the authorities of DEPT547 user profile. Because the group profile is the owner of PAY1, all members of the group have authority to PAY1 except SWENSONS. SWENSONS cannot use PAY1 because *EXCLUDE authority was specified for SWENSONS. Similarly, because the DEPT547 profile has authority to PAY2, members of the group have the same authority. However, because the group does not own PAY3 and does not have authority specifically given for it, JONESP, SWENSONS, and DEPT547 do not have authority to PAY3 (unless public authority is other than *EXCLUDE). However, MILLERK, who is a member of the group, can use PAY3 because he was specifically given authority to it.

DEPT547 can also start a job, although normally the group would be used only for authority and ownership control. If you do not want a user to sign on as DEPT547, specify PASSWORD(*NONE) on the group profile.

Method 2: If an existing user profile such as MILLERK already contains the authorities needed for a department, it can be made a group profile. No change needs to be made to the MILLERK user profile. The other profiles that want to use the existing user profile would specify the MILLERK user profile as the group in the group profile (GRPPRF) parameter.

Ownership and authorities to any objects created by members of the group should be considered as discussed in the topic "Group Ownership of Objects" on page 4-11.

A disadvantage of this method is that the MILLERK user profile could not have any private authority to an object because each member would automatically be given the same authority to the object.

Another disadvantage of using method 2 is that the name does not follow the recommended naming conventions of a group profile or department group profile. When displaying authority to objects, it is not apparent that the MILLERK user profile is a group profile.

A better approach is to create a department group, as mentioned under method 1. You can transfer ownership of objects from the MILLERK user profile to the department group profile, and then make the MILLERK user profile a member of

the department group, or you can use the Grant User Authority (GRTUSRAUT) command to give the department group the same authorities as the MILLERK user profile by using the MILLERK user profile as a reference for authority. For more information about the Grant User Authority command, see the topic "Grant User Authority."

If a group of programmers shares common tasks, it may be desirable to include the programmers as members of a group. One approach is to make the IBM-supplied QPGMR user profile the group profile. Because QPGMR may contain authorities that should not be used by all programmers, another approach is to create a programmer group profile that contains only the authorities needed for the required functions. This may be desirable if QPGMR is the owner of all programs that are used in an application program and the programmer who developed the application program does not own the application program.

Security Consideration

For end-user tasks, it is normally not desirable to use an IBM-supplied user profile, such as QPGMR, as a group profile. Using the authority of the group profile also includes any special authorities associated with the IBM-supplied user profile. If QPGMR is used as a group profile, members of the group have job control special authority and can cancel any job on the system.

Grant User Authority

The Grant User Authority (GRTUSRAUT) command copies authorities from one user profile to another.

Security Consideration

The Grant User Authority (GRTUSRAUT) command should only be used in special cases because it can take a long time to run and the time to save the system also increases. Authorization lists and group profiles are better, more effective methods for shared authority for objects.

For example, if a new user profile WILSON is created and it needs the same authorities as DEPT547 (Figure 4-3 on page 4-13), the GRTUSRAUT command can be used to specifically add authority to the WILSON profile for PAY1 and PAY2 even though WILSON is not a member of the group. However, if another object is subsequently created and both DEPT547 and user profile WILSON need authority to it, then both profiles must be specifically given authority to it.

In contrast, the group profile function allows both the existing user profiles (such as MILLERK, JONESP, and SWENSONS) and the new user profile (WILSON) to become part of the same group (DEPT547). Then, if any changes are needed to authorities that affect all members of the group, only the group profile needs to be changed.

Similarly, an authorization list (Figure 4-1 on page 4-8) allows both the existing user profiles (such as TERRY, JUDY, SCOTT, KARENS and MARY) and the new user profile (BETTY) to become part of the same authorization list (AUTL1) by adding BETTY to the list. If any changes are needed to the authorities that affect two user profiles on the list (each with a different authority) and several objects, only the authorities specified on the authorization list need to be changed.

The GRTUSRAUT command actually causes specific authorities to be added to the user profile, and each authority that is copied becomes part of the user profile. These authorities can be displayed using the Display User Profile (DSPUSRPRF) command. In contrast, the group profile function does not copy the separate authorities to each member of the group. The group profile function can significantly reduce the number of private authorities that exist on the system thus improving save system time.

Programs That Adopt the Owner's Authority

When you create a program that will be used by others, you must give each user authority not only to the program but also to the objects (such as files) used by the program. However, you can specify when the program is created, and whether the program is to always run under the authority of the program owner. Users of the program do not need authority given specifically to them for the objects, because they adopt the owner's authority. The owner of the program needs authority to the objects. Users have authority for the objects used by the program only when they are running the program and subsequent programs called by the program.

Programs that adopt authority obtain the authority from the owner of the program while the program is active in the job. A program is active (exists in the program stack) when it has been called by the CALL or TFRCTL command.

To specify that a program is to run under the owner's user profile, you must specify the following parameter and value on a create program command:

```
USRPRF(*OWNER)
```

You can find out if a program has used this parameter by using the Display Program (DSPPGM) command.

When a program adopts the owner's authority, users of the program are given their own authority and the private authorities are given specifically to the owner of the program. In addition to the private authorities of the owner, the special authorities specified in the owner's user profile are adopted.

If the owner of the program has a group profile, no authorities of the program owner's group profile are used. If the owner of the program has been given *EXCLUDE authority to an object as a private authority, the private authority is ignored when authority checking is done for the program adopted authority.

The following figure shows a typical use of the adopt function:

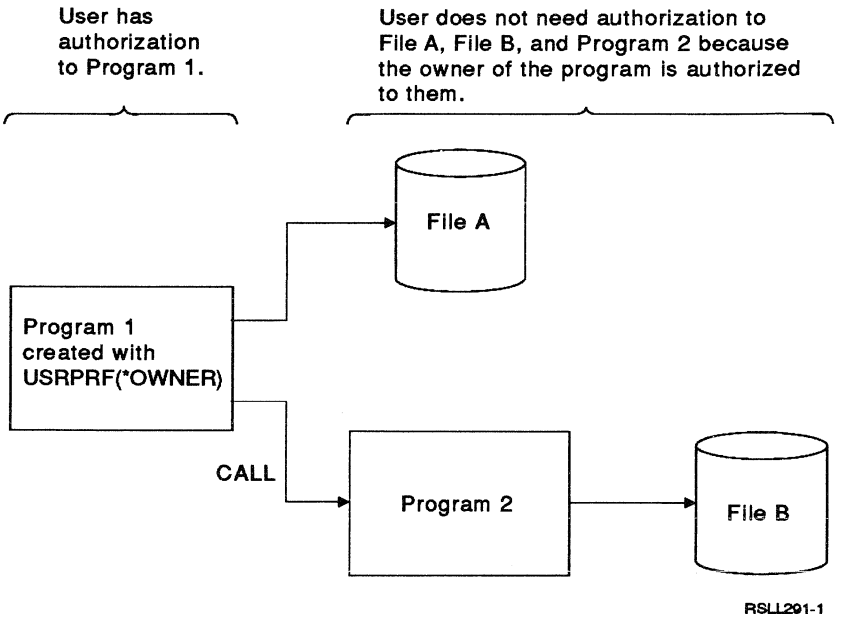


Figure 4-4. Program Adopt Function

When a program is running using the owner's authority, it is possible for a library to be added to the library list for which the current user of the program does not have authority. The current user keeps authority to that library while the library is in the library list. When a program is using adopted authority, you should ensure that any private libraries added to the library list are removed before exiting the program. To ensure the security of a program that uses adopted authority, see the topic "Library List Considerations" on page 5-1.

Security Risk

Allowing a program to run under the owner's user profile is an intentional release of control (authority), which may allow unanticipated access to objects.

Programs that adopt authority work well when all the objects in an application program are owned by and authorized to the program owner's user profile. Any objects created while a user is running a program under the owner's user profile are owned by the user profile running the program, not by the user profile that owns the program.

If a program running under the owner's user profile transfers control to another program with the Transfer Control (TFRCTL) command, all the adopted authorities for the program that transferred control are not adopted. See Figure 4-5.

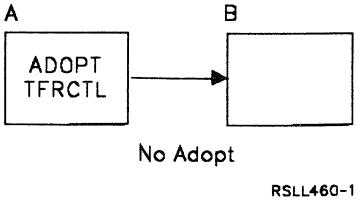


Figure 4-5. Program Adopted Authority Is Not Transferred

However, if the program running under an owner's user profile calls another program by means of a CALL command, authority is adopted by the program that was called. See Figure 4-6.

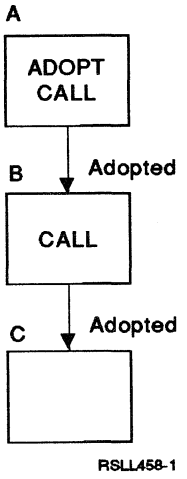


Figure 4-6. Program Adopted Authority Is Transferred

Furthermore, authority continues to be adopted by other programs from the program that was called, even if control is transferred by the program that was called. See Figure 4-7.

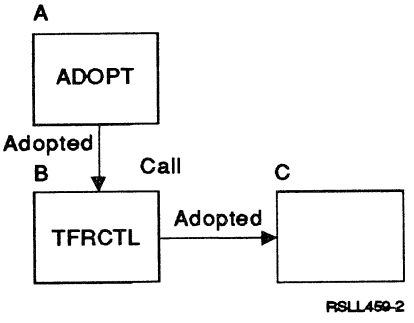


Figure 4-7. Program Adopted Authority in Program Stack Is Transferred

As long as the program running with the adopted authority remains in the program stack, any programs that are lower in the stack use the adopted authority. Several programs in the program stack may use adopted authority. Adopted authority checking is always additive (that is, it includes all the authority of the owners of programs using adopted authority in addition to the user's own authority).

One exception in which adopted authority is different is when *EXCLUDE authority is specified. For example, program PGMA adopts the owner's authority of *OBJMGT and PGMA calls PGMB. Assume the user running PGMA has *EXCLUDE authority to PGMB. The user running PGMA has *OBJMGT authority for PGMB even though his private authority is *EXCLUDE. Also, if the owner's authority for PGMB is *EXCLUDE and the user's authority to PGMB is *OBJMGT, the user gets *OBJMGT authority to PGMB.

If your application design includes an initial program that displays a menu, and all functions of the menu are done by CALL commands to other programs, you

can simply create the initial program as USRPRF(*OWNER) and all application functions operate under the owner's profile.

Assume you have a program that displays a menu that offers several application program choices, one of which is the use of a query program. Specifying adopted authority at the menu program level allows the query program user to access the same files as the user who created the program.

If you do not want the user to run the query program under the owner's authority, an alternative is to use adopted authority, not on the menu program but instead, in the programs or menu programs called from the menu. This approach uses the adopted authority in addition to the user's own authority only for those programs that adopt the owner's authority. By not specifying adopted authority for the query program, only the user's authority to the query program is used.

If you do not want the primary program owner's authority added to the secondary program owner's authority, a second alternative is to have the program ignore the adopted authority. See "Programs That Ignore Adopted Authority" on page 4-18 for information about ignoring program adopted authority.

If a primary program that uses adopted authority calls a secondary program, the secondary program still operates under the owner's authority of the primary program. If the secondary program has a different owner than the primary program and USRPRF(*OWNER) is specified for the secondary program, then the owner's authority of the secondary program is added to the owner's authority of the primary program as long as the program is active.

Several functions may interrupt a job that is running. If a program in the program stack uses adopted authority, none of the program owner's authority is used as a source of authority while using the following:

- System request
- Attn key (If a Transfer to Group Job (TFRGRPJOB) command is running, adopted authority is not passed to the group job.)
- Break-message-handling program
- Debug functions

Only the owner, the security officer user profile, or a user with all object (*ALLOBJ) and security administrator (*SECADM) special authorities can transfer ownership of a program that runs under the owner's user profile.

The program adopt function is not used when a change occurs to the job queue or output queue parameters on the Change Job (CHGJOB) command. The user profile must have authority to the queue to change these parameters.

Programs That Ignore Adopted Authority

In some cases, it may be desirable (such as in menu security) to ignore the adopted authority from any previous programs in the program stack. To specify that a program ignore adopted authority from any previously called program in the program stack, use the following parameter and value on the Change Program (CHGPGM) command:

```
USEADPAUT(*NO)
```

The value USEADPAUT(*NO), allows the program to run without using the authority from a previous program in the program stack that adopted the owner's user profile.

Normally, all programs in the program stack will use the authority from the program that adopts the owner's authority while the program is active in the job. The use adopted authority (USEADPAUT) parameter on the Change Program (CHGPGM) command allows you to specify that a program will either use or ignore the adopted authority from previous programs in the program stack.

Assume you have created a program that displays a menu and that all options on the menu call other programs by the CALL command. If you want all options to run under your authority, you can simply specify USRPRF(*OWNER) for the menu program. All menu options will operate under your user profile because the USEADPAUT parameter for all programs being called default to *YES. In the following example, the differences between USEADPAUT(*YES) and USEADPAUT(*NO) are shown.

Assume USERB has created a program called STARTPGM that displays a menu. An option from the menu calls the user QUERYPGM program owned by USERC. The STARTPGM program uses programs and files that USERB owns. If adopted authority is specified at the STARTPGM program level, USERA, accessing the QUERYPGM program, can add, change, or delete the records associated with the STARTPGM program by using the USERB s adopted authority. See Figure 4-8.

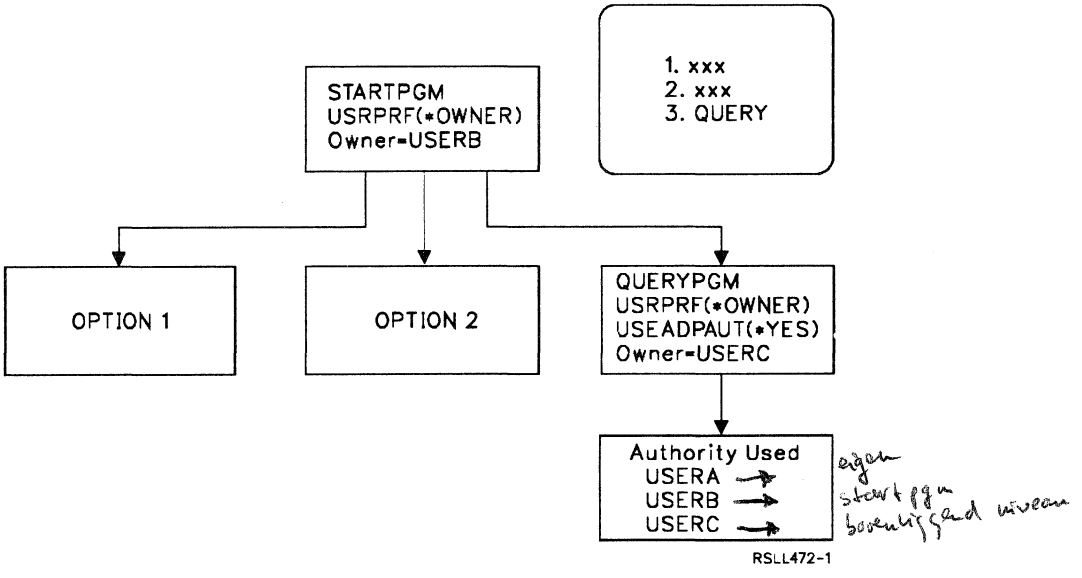


*Figure 4-8. Program Adopt with USEADPAUT(*YES) Specified*

If it is not desirable to have USERA accessing the files with USERB's authority when running the QUERYPGM program, an alternative is to have USERC change the QUERYPGM program by specifying USEADPAUT(*NO) on the Change Program (CHGPGM) command. In this case, USERB's authority from program STARTPGM is not adopted by USERA when running the QUERYPGM program. The QUERYPGM program will ignore USERB's authority for program STARTPGM. See Figure 4-9 on page 4-20.

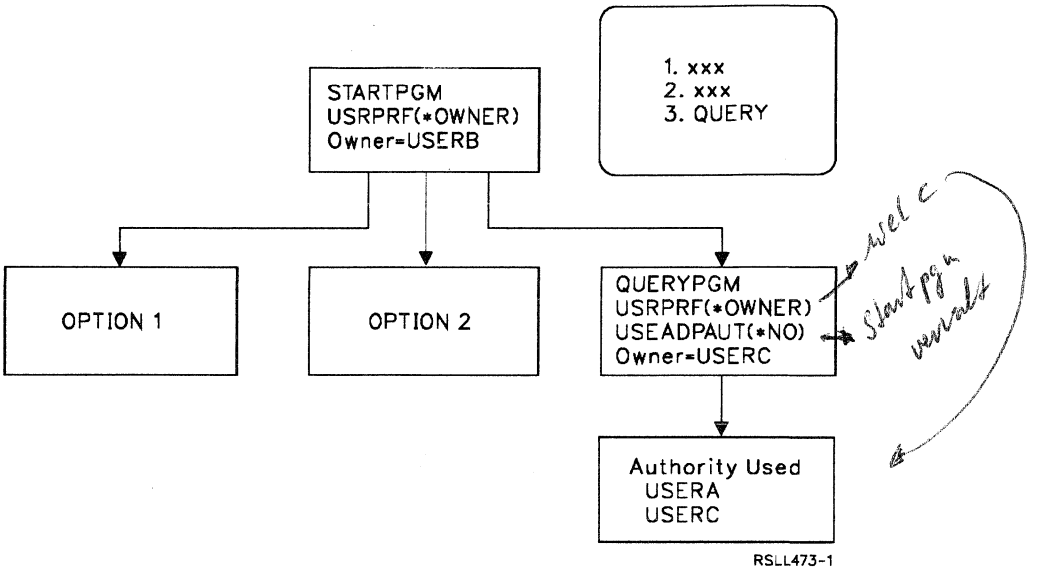


*Figure 4-9. Program Adopt with USER(*OWNER) USEADPAUT(*NO) Specified*

However, if the QUERYPGM program specifies USRPRF(*USER) USEADPAUT(*NO), then USERB's authority for the STARTPGM program and USERC's authority for the QUERYPGM program are not added to USERA's authority. Only USERA's authority to the QUERYPGM program is used. See Figure 4-10.

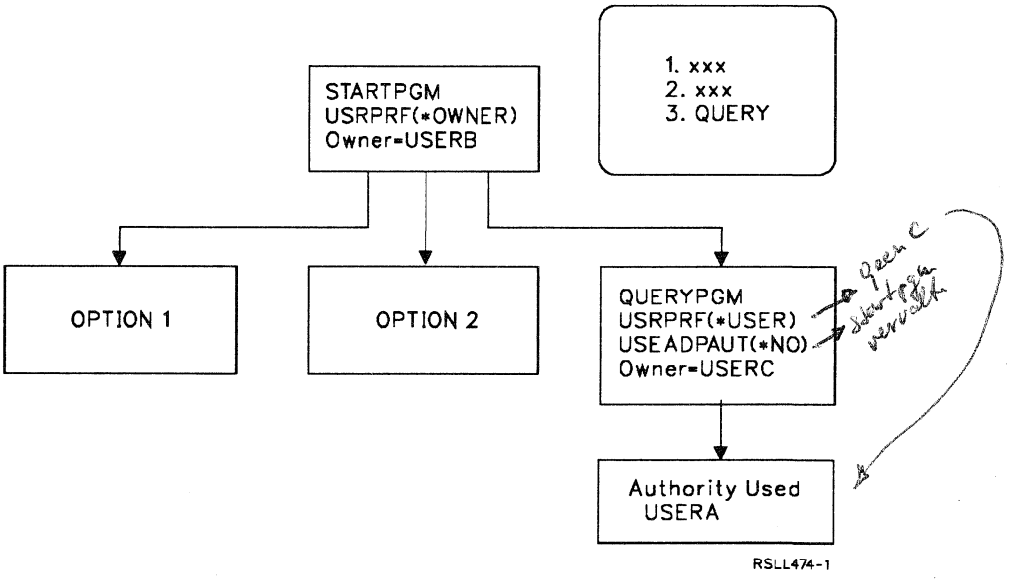


*Figure 4-10. Program Adopt with USER (*USER) USEADPAUT(*NO) Specified*

Considerations for Programs that Adopted Authority

Program adopt considerations that apply to the USEADPAUT parameter follow:

- The use adopted authority parameter (USEADPAUT(*YES or *NO)) can be specified only on the Change Program (CHGPGM) command.
- If a program is created again using REPLACE(*YES) from a create program (CRTxxxPGM) command (for example, CRTRPGPGM or CRTCLPGM), the

new copy of the program will use the value for the USRPRF and USEADPAUT parameters from the replaced program.

- Only the owner or a user profile with *ALLOBJ and *SECADM special authorities can change the value of the user profile parameter.
- Only the owner or a user profile with *ALLOBJ and *SECADM special authorities can change the value of the use adopt parameter.

Program adopt considerations for authority if USRPRF(*OWNER) USEADPAUT(*YES) is specified.

- The adopt function is additive for all programs in the program stack if USRPRF(*OWNER) USEADPAUT(*YES) is specified. For example, if a primary program adopts the owner's authority, any secondary programs that are created with USRPRF(*USER) still operate under the owner's authority of the primary program.
- A program using adopted authority operates under the owner's authority in addition to the user's authority. If the user has authority and the program owner is excluded, access is allowed.
- If a program that uses adopted authority submits a job, that submitted job does not have the adopted authority of the submitting program.
- If the job is running with program adopted authority, and the owner of the program is a member of a group profile, the authority of the owner's group profile is not used. Only the authority of the program's owner is used.

  See the topic "Submitting Jobs That Adopt Authority" on page 5-15 for a discussion of a technique to submit jobs that adopt authority.

Other considerations for using the program adopt function follow:

- If someone other than the program's owner or a user with *ALLOBJ and *SECADM special authorities restores a program that runs under the adopted authority of its owner, the program is restored to the owner. All other users' specific and the public authorities to the restored program are revoked in order to prevent a possible security exposure.
- The adopt function remains in effect as long as the program remains in the program stack and subsequent programs do not specify USEADPAUT(*NO).
- Any objects created are owned by the user of the program or by the user's group profile, not by the owner of the program.
- The only attributes of the owner's user profile that are adopted are:
  - All special authorities
  - All private authorities

For information about displaying programs that adopt the owner's authority, see the topic "Displaying Programs That Adopt" on page 8-41.

Program Adopt Considerations for Group Profiles

When a job is running, sources of authority can differ. For example, User A is a member of Group 1 (see Figure 4-11 on page 4-22). User B is a member of Group 2 and owns program PGMX, which is specified to run under the owner's user profile (USRPRF(*OWNER)). If User A calls program PGMX, the authority used when running the program is the authority of User A, Group 1, and User B. The authority of Group 2 is ignored. See Figure 4-11 on page 4-22.

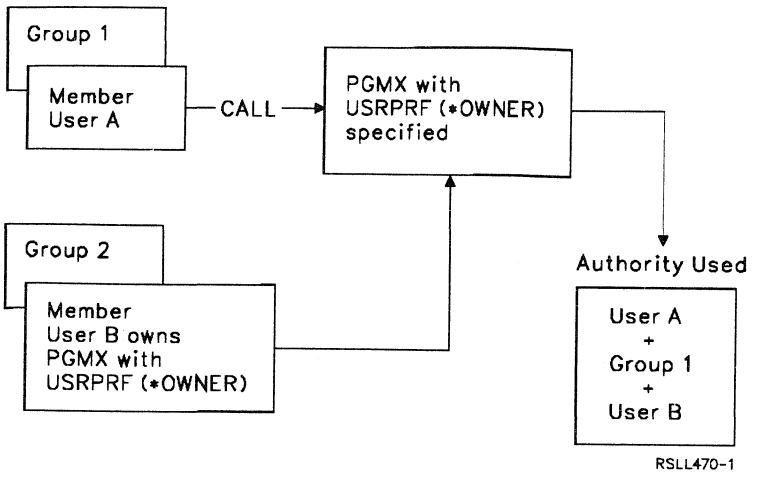


Figure 4-11. Sources of Authority When Running a Program

Default Public Authority for Newly-Created Objects

When objects are created into a library, the public authority for the object will, by default, be set by using the CRTAUT value of the library. By specifying:

```
CRTLIB LIB(TESTLIB) CRTAUT(*USE) AUT(*LIBCRTAUT)
```

the library TESTLIB is created. All objects created into library TESTLIB will, by default, have public authority of *USE. The public authority for library TESTLIB is determined by the CRTAUT value of library QSYS.

By specifying:

```
CRTDTAARA DTAARA(TESTLIB/DTA1) TYPE(*CHAR) +
          AUT(*LIBCRTAUT)

CRTDTAARA DTAARA(TESTLIB/DTA2) TYPE(*CHAR) +
          AUT(*EXCLUDE)
```

data area DTA1 is created into library TESTLIB. The public authority of DTA1 is *USE based on the CRTAUT value of library TESTLIB.

Data area DTA2 is created into library TESTLIB. The public authority of DTA2 is *EXCLUDE. *EXCLUDE was specified on the AUT parameter of the Create Data Area (CRTDTAARA) command.

An authorization list can also be used to secure an object when it is created into a library. By specifying:

```
CRTAUTL AUTL(PAYROLL)
CRTLIB  LIB(PAYLIB) CRTAUT(PAYROLL) +
        AUT(*EXCLUDE)
```

an authorization list called PAYROLL is created. Library PAYLIB is created with the public authority of *EXCLUDE. By default, an object created into library PAYLIB is secured by authorization list PAYROLL.

By specifying:

```
CRTPF FILE(PAYLIB/PAYFILE) +
      AUT(*LIBCRTAUT)

CRTPF FILE(PAYLIB/PAYACC) +
      AUT(*CHANGE)
```

file PAYFILE is created into library PAYLIB. File PAYFILE is secured by authorization list PAYROLL. The public authority of file PAYFILE is set to *AUTL as part of the Create Physical File (CRTPF) command. *AUTL indicates that the public authority for file PAYFILE is taken from the authorization list securing file PAYFILE, which is authorization list PAYROLL.

File PAYACC is created into library PAYLIB. The public authority for file PAYACC is *CHANGE since it was specified on the AUT parameter of the CRTPF command.

Notes:

1. The *LIBCRTAUT value of the AUT parameter that exists on most CRT commands indicates that the public authority for the object is set to the CRTAUT value of the library that the object is being created into.
2. The CRTAUT value on the library specifies the default authority for public use of the objects created into the library. These possible values are:

***SYSVAL**	The public authority for the object being created will be the value specified in system value QCRTAUT
***ALL**	All public authorities
***CHANGE**	Change authority
***USE**	Use authority
***EXCLUDE**	Exclude authority
authorization list name	The authorization list secures the object

3. The CRTAUT value of the library is not used during a move (MOVOBJ), create duplicate (CRTDUPOBJ), or restore of an object into the library. The public authority of the existing object is used.
4. If the REPLACE(*YES) parameter is used on the create command, then the authority of the existing object is used instead of the CRTAUT value of the library.

Specifying Authority for Objects

Authority for objects can be specified by using the following commands:

1. Edit Object Authority (EDTOBJAUT) command. Both object authority and data authority can be managed interactively through the use of the EDTOBJAUT command. This command shows which users have authority to the object and what their authority is for the object. You can change or remove existing authorities or add new users with this command.
2. Grant Object Authority (GRTOBJAUT) command. The GRTOBJAUT command can be used in batch or interactively to change existing authorities or add new users.

3. Revoke Object Authority (RVKOBJAUT) command. The RVKOBJAUT command can be used in batch or interactively to remove existing authorities.

You cannot use the Grant and Revoke Object Authority commands if the object type is an authorization list (*AUTL). You must use the Edit Authorization List (EDTAUTL) command to add, change, and remove users, or use the Add Authorization List Entry (ADDAUTLE), Change Authorization List Entry (CHGAUTLE), or Remove Authorization List Entry (RMVAUTLE) command to add, change, or remove users on an authorization list.

All types of authority can be specified for all types of objects with these commands except for the following:

- Data authority cannot be specified for logical files.
- No authority can be specified for filed documents and folders. Documents and folders have their own security commands. For more information about these commands, see "Working with Document Library Objects" on page A-4.

To grant object and data authority, the user must be one of the following:

- The security officer
- A user with all object (*ALLOBJ) special authority
- The owner of the object
- A user with object management (*OBJMGT) authority or, if the object is an authorization list, a user with authorization list management (*AUTLMGT) authority and any other authorities that are being granted

The user must also have any authorities being granted except for *EXCLUDE authority. Any user who can grant or revoke authority can grant or revoke *EXCLUDE. Only the security officer, a user with all object (*ALLOBJ) special authority, or the object owner can grant object management authority and authorization list management authority. If the object owner is a group profile, then each member (if the member has no authority given specifically for the object) associated with the group profile acts as the object owner and can grant object management authority or authorization list management authority.

You can give authority to a group of objects by specifying a generic name that the group of objects satisfies. You specify the generic name by typing the beginning character string that each object has in common and following it with an asterisk (*). In the following example, users RSMITH, RJONES, TBROWN, and WDOUGLAS have authority to use the objects whose names start with ORD in the library DSTPRODLB. The objects are all of the same type (program), and the users are given object operational and read authorities (*USE in the AUT parameter), which means they can run the programs.

```
GRTOBJAUT   OBJ(DSTPRODLB/ORD*) OBJTYPE(*PGM)
            USER(RSMITH RJONES TBROWN WDOUGLAS)
            AUT(*USE)
```

You can also grant authority specifically to all objects of all types in a library by specifying the command as follows:

```
GRTOBJAUT   OBJ(DSTPRODLB/*ALL) OBJTYPE(*ALL)
            USER(RSMITH RJONES TBROWN WDOUGLAS)
            AUT(*USE)
```

Security Consideration

The generic grant functions can take a significant amount of time to run, depending on the number of objects that are selected. When a generic grant function is requested, a specific library must be named.

Granting authority generically applies to objects that exist at the time of the generic grant. If a new object that matches the criteria is created later, the generic grant does not apply.

Instead of specifically granting the users authority, the recommended procedure is to create an authorization list for users RSMITH, RJONES, TBROWN, and WDOUGLAS. You can then use the generic grant function to grant the authorization list authority to the objects.

```
GRTOBJAUT   OBJ(DSTPRODLB/ORD*) OBJTYPE(*ALL)
            AUTL(AUTLIST)
```

The advantages of using this approach are to make it easier to add and remove other users for the same objects and to reduce the number of authorities on the system to improve the time it takes to save the system.

When you create most objects, you can specify the public authority for an object on the AUT parameter of the created object, or the object uses the default for the AUT parameter. **Public authority** is the authority granted to users who have no other source of authority. Public authority can be specified as:

- All: Every user can use the object like the owner except he cannot perform those functions restricted specifically to the owner.
- Change: Every user can change and use the object.
- Use: All users have some authority to use the object.
- Exclude: Only the owner, security officer, someone with all object (*ALLOBJ) special authority, and other users who have authority given specifically for the object can use it.

For a more detailed description of change, all, use, and exclude authority, see the topic "Specific Authority" on page 4-1.

When authority has been granted for an object, it can only be removed using the Edit Object Authority or Revoke Object Authority command.

Notes:

1. You cannot use the Grant Object Authority (GRTOBJAUT) command for documents. See the topic "Working with Document Library Objects" on page A-4 for more information about the commands used for documents or folders.
2. You cannot use the Grant Object Authority (GRTOBJAUT) command for X authorization lists. You must use the Edit Authorization List (EDTAUTL) command, the Add Authorization List Entry (ADDAUTLE) command, the Change Authorization List Entry (CHGAUTLE) command, or the Remove Authorization List Entry (RMVAUTLE) command to give a user authority on an authorization list.

Appendix D, "Authority Required for Objects Used by Commands" shows you what authorities are needed, in addition to command authorities, to use a command on an object. For example, if you create a data area and you want a user or the public to be able to change it, the Data Areas table (showing the

Change Data Area (CHGDTAARA) command) in Appendix D, "Authority Required for Objects Used by Commands" tells you that object operational (*OBJOPR) and update *(UPD) authorities are required.

If you create a data area with AUT(*EXCLUDE) specified, the table in Appendix D, "Authority Required for Objects Used by Commands" tells you that you must grant authority to users that need to update the data area. This can be done by specifying AUT(*READ *UPD) on the Grant Object Authority command for the required users.

Because both libraries and objects being used default to *CHANGE public authority or are owned by you, you should usually have the authorities required for most functions. If you want to do functions on objects that you do not own or have authority to, you must get authority from the owner.

When a physical file is created using the following command, the AUT parameter gives change (*CHANGE) authority to all users who do not have authority given specifically to them for the physical file ORDHDRP:

```
CRTPF FILE(ORDHDRP) SRCFILE(QDDSSRC) AUT(*CHANGE)
```

This means that all users on the system who do not have authority given specifically for the file, have object operational authority (the right to open the file and look at its description), as well as read, add, update, and delete data authorities to the file.

Removing Authority for Objects

You can remove a user's authority to an object. The following list shows which commands can be used to remove authority.

- Edit Object Authority (EDTOBJAUT) command (interactive)
- Revoke Object Authority (RVKOBJAUT) command (batch and interactive)
- Change Authorization List Entry (CHGAUTLE) command (batch and interactive)
- Edit Authorization List (EDTAUTL) command (interactive)
- Remove Authorization List Entry (RMVAUTLE) command (batch and interactive)

Security Risk

Removing all of a user's authority for an object may result in the user having more authority to the object than he did before. For example, if the user has *EXCLUDE authority to an object, and he is on an authorization list with public authority specified as *CHANGE, removing exclude authority to the object will give him *CHANGE authority. If the public authority for an object is more than the authority given specifically to the user, removing the authority given specifically to the users will result in the user having more authority than he did before.

In the following example, the Edit Object Authority command shows the Edit Object Authority display. This display allows you to display, add, change, or remove authority for an object. For example:

```
EDTOBJAUT OBJ(ORDLIB/ORDMNU) TYPE(*PGM)
```

The following display is shown:

```
                    Edit Object Authority

Object . . . . . .:  ORDMNU          Object type . . . . : *PGM
  Library . . . . . :  ORDLIB        Owner . . . . . . . : THUMBA

Object secured by an authorization list . . . . . .: *NONE

Type changes to current authorities, press Enter.

             Object
User         Authority
THUMBA       *ALL
MILLERK      USER DEF
WAYS         USER DEF
GROUP547     *ALL
*PUBLIC      *USE

                                                              Bottom

F3=Exit  F5=Refresh  F6=Add new users  F10=Grant with reference object
F11=Display detail  F12=Cancel  F17=Top  F18=Bottom
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Pressing F11 (Display detail) displays the specific authorities for the users.

The following display is shown:

```
                    Edit Object Authority

Object . . . . . .:  ORDMNU          Object type . . . . : *PGM
  Library . . . . . :  ORDLIB        Owner . . . . . . . : THUMBA

Object secured by an authorization list . . . . . .: *NONE

Type changes to current authorities, press Enter.

             Object     -----Object-------   ----------Data-------------
User         Authority  Opr   Mgt   Exist    Read  Add   Update  Delete
THUMBA       *ALL        X     X      X       X     X       X       X
MILLERK      USER DEF    X     X              X
WAYS         USER DEF    X     X              X
GROUP547     *ALL        X     X      X       X     X       X       X
*PUBLIC      *USE        X                    X

                                                              Bottom

F3=Exit  F5=Refresh  F6=Add new users  F10=Grant with reference object
F11=Non-Display detail  F12=Cancel  F17=Top  F18=Bottom
```

You add or change users' authorities from this display by typing an X in one of the authority fields, or you can remove users' authorities by replacing the X with a space. To add a user to the list of users, press F6, enter a user name in the *User* column, and place an X in the authority fields you want for this user.

The EDTOBJAUT command does not allow you to remove authority for a group of objects. You can use the Revoke Object Authority (RVKOBJAUT) command to remove authority for a group of objects by specifying a generic name that the group of objects satisfies. You specify the generic name by typing the starting character string that each object has in common and following it with an asterisk (*).

In the following example, using the Revoke Object Authority command, all the authority RSMITH has for all objects (of the type program) whose names start with ORD in the library DSTPRODLB is removed:

```
RVKOBJAUT   OBJ(DSTPRODLB/ORD*) OBJTYPE(*PGM)
            USER(RSMITH) AUT(*ALL)
```

You can also remove all the authority of one or more users to all objects of all types in a library by specifying the following:

```
RVKOBJAUT   OBJ(DSTPRODLB/*ALL)   OBJTYPE(*ALL)
            USER(RSMITH JONESP TBROWN WDOUGLAS)
            AUT(*ALL)
```

Security Consideration

The generic revoke functions can take a significant amount of time to run, depending on the number of objects that are selected. When a generic revoke function is requested, a specific library must be named.

Revoking authority generically applies to objects that exist at the time of the generic revoke. If a new object that matches the criteria is created later, the generic revoke does not apply.

All authorities can be removed from a user by the security officer, the object's owner, or a user who has all object (*ALLOBJ) special authority, using the Revoke Object Authority (RVKOBJAUT) command except for object types *DOC, *FLR, or *AUTL. See "Working with Authorization Lists" on page A-1 and "Working with Document Library Objects" on page A-4 for the commands used to revoke authority for these object types.

If the object owner is a group profile, then each member (if the member does not have authority specifically given) associated with the group profile shares the group profile's authority and can remove the specific authorities for the objects.

Security Consideration

When revoking authority to IBM-supplied commands in library QSYS, you should also revoke authority to any similar commands in library QSYS38.

Giving Authorization Other Than Public Authorization

Sometimes, a user may have to perform a function requiring more than the public authority. For example, a menu option may require a file member to be cleared or a member to be added or removed. These functions require more authority than those provided by system defaults. You can give this authority by doing one of the following:

- Authorize a particular user to a specific object.
- Create an initial program that adopts the owner's profile. This assumes that the objects are owned by the same user profile. If different parts of an application are owned by different user profiles, the high-level language program for each part of an application can be created in this manner. If a security-sensitive function must be done in batch mode, the high-level language batch program can be created with the adopt function.
- Use a special control language (CL) program for the required function and adopt the owner's profile for authorization.

- Authorize the required function of the object for public use. For example, to clear a member requires object management authority. You can give this authority to the public for certain objects that are not sensitive, such as work files.
- Authorize the object to an authorization list so the authority for the object comes from the authorization list.
- Specify an authorization list for files and libraries when you create them.

Authority Checking

Authority checking is a function done by the system to verify authority to an object. When an authority is found, authority checking stops. The authority found is used for the object. Authority is not additive except for the program adopt function. The program adopt function uses the authority of the program's owner in addition to the user's own authority.

The system verifies a user's authority to an object in the following order:

1. User's all object (*ALLOBJ) special authority
2. User's specific authority to an object
3. User's authority on the authorization list (if one is specified for the object)
4. User's group all object (*ALLOBJ) special authority for the object (if the user is a member of a group)
5. User's group authority to the object (if the user is a member of a group)
6. User's group authority on the authorization list (if there is one specified for the object)
7. Public authority specified for the object
8. Public authority specified for the object in the authorization list (if the public authority in the object is specified as *AUTL)
9. Adopted authority is added to any authority found

The following figure shows the order in which the system verifies authority.

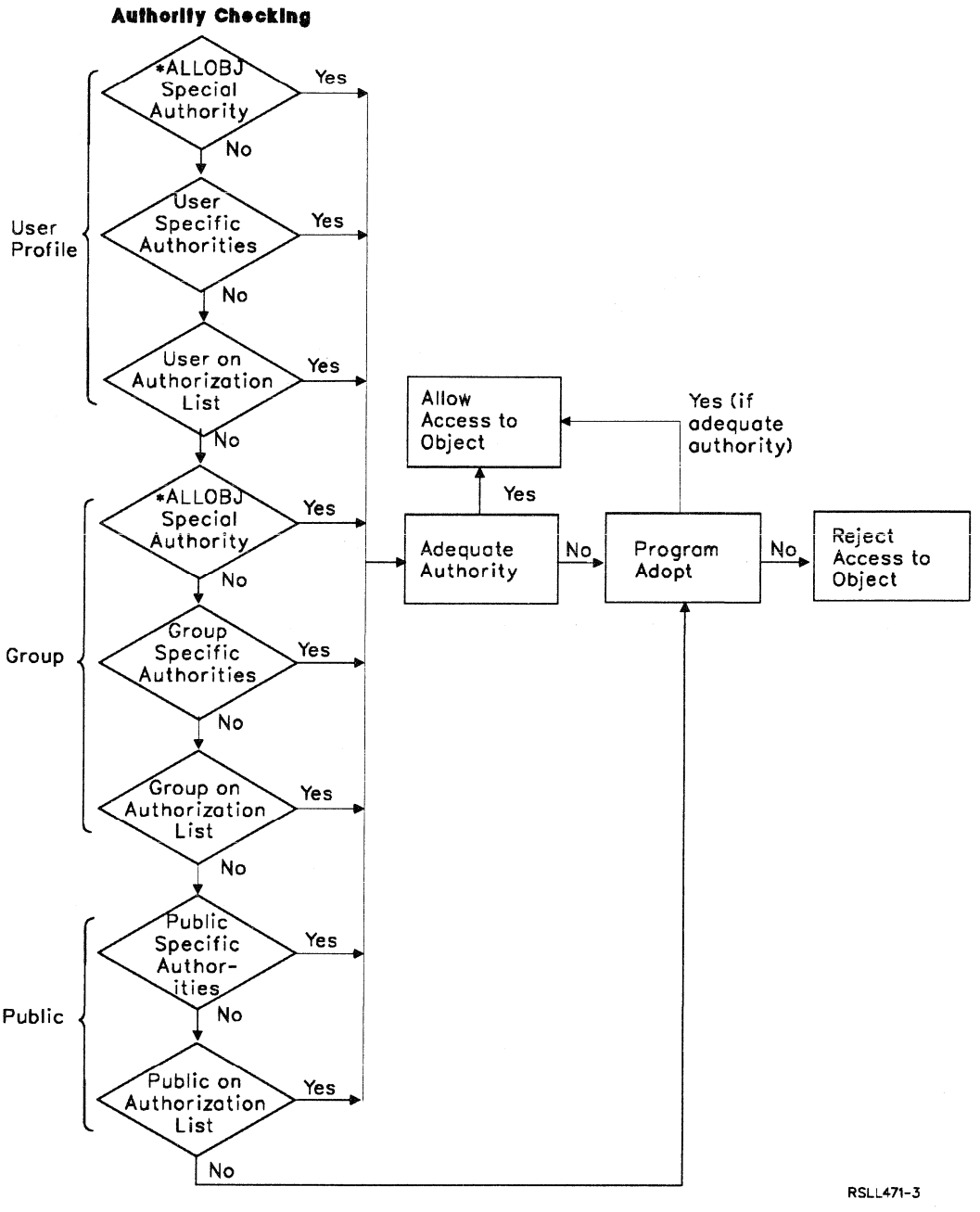


Figure 4-12. Authority Checking

System Performance Considerations

Because the system does authority checking on an object each time it is accessed, the best performance can be achieved by using the public authority for the object and granting **no** private authorities.

If your security strategy requires the use of private authorities, try to keep the private authority greater than what is specified for the public authority for better performance.

If a large number of private authorities are used, management of these authorities becomes more difficult and the time it takes to save the system (SAVSYS command) and to restore authority (RSTAUT command when restoring the system) increases.

Authority Holders

Authority holders, found in library QSYS, allow the authority for program-described database files to be kept by the system even when the file does not exist. For example, this allows the system to keep the authority for System/36 environment applications that often delete program-described files and then re-create them.

Creating Authority Holders

Authority holders can be created for files that already exist, or can be created to reserve the name of a file and control who can create that file before the file exists.

For example, user GLORIA owns a payroll program that often deletes and re-creates the file GENPAY. Another user TOM uses the file GENPAY in another program. Each time GENPAY is deleted and re-created, GLORIA has to specifically give TOM authority to the file.

By creating an authority holder (see Figure 4-13) for the file GENPAY, GLORIA does not need to authorize TOM each time the file is deleted and then re-created. When the file is re-created, it is automatically linked to the authority holder. An authority holder must have the same name as the file.

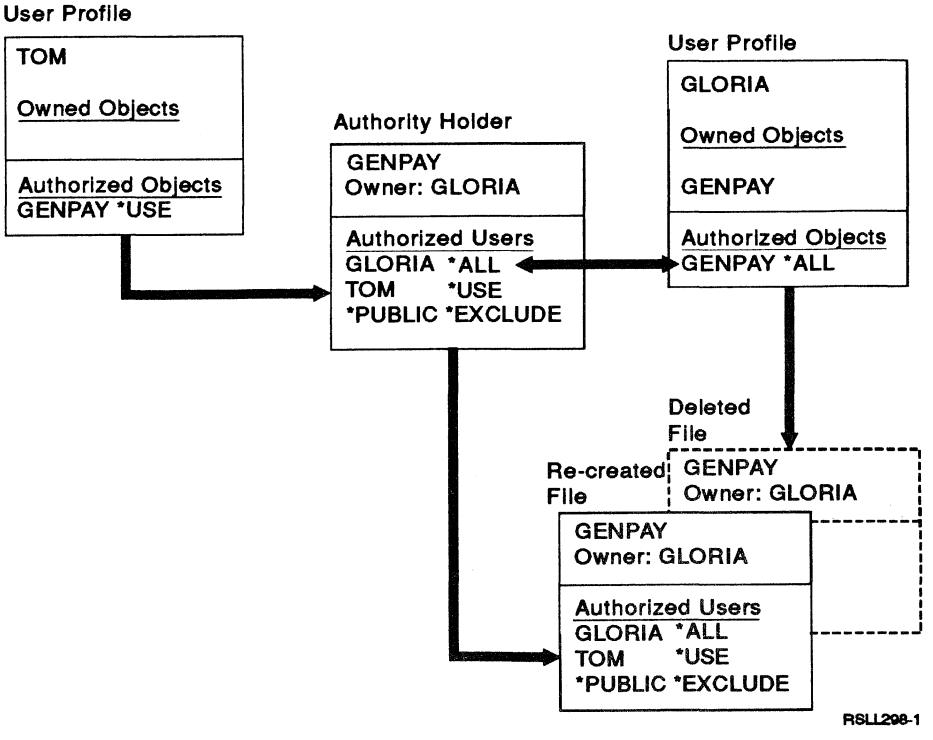


Figure 4-13. Example of an Authority Holder

When creating authority holders for files that already exist, consider the following:

- The user creating the authority holder must have *ALL authority to the object.
- The existing object's authority is copied to the authority holder.
- The owner of the object becomes the owner of the authority holder regardless of the user creating the authority holder.
- The public authority for the authority holder comes from the public authority of the object.
- The authority holder is limited to a program-described database file.

When authority holders are deleted, all authorities that exist for the authority holder are granted to the file if the file exists.

The Create Authority Holder (CRTAUTHLR) command is shipped with the public authority *EXCLUDE. Only users with all object (*ALLOBJ) special authority can use this command unless granted authorization.

Authority Holder Considerations

Security Risk

Care should be taken when giving other users authority to the Create Authority Holder (CRTAUTHLR) command because a user can become the owner of an object that he has no authority for.

For example, user TOM creates an authority holder named PAY1. Tom gives himself and user GLORIA *ALL authority and specifies *EXCLUDE authority for the public.

GLORIA owns a file named PAYACCT. No authority has been given specifically to TOM for the file PAYACCT and the public has no authority (*EXCLUDE) to it. See Figure 4-14.

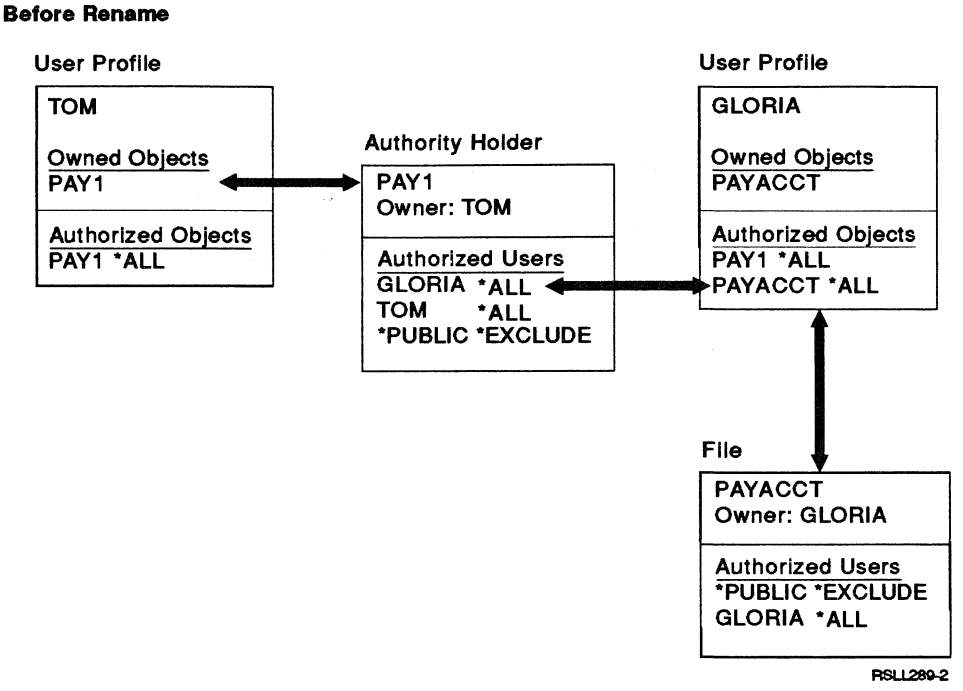


Figure 4-14. Example of an Authority Holder Before Rename

If GLORIA renames PAYACCT to PAY1, TOM now has *ALL authority to the renamed file PAY1 and becomes the owner when before he had no authority to it. See Figure 4-15 on page 4-34.

The same situation can happen if GLORIA moves or restores PAY1 to a different library. However, GLORIA is notified that PAY1 has been linked to an authority holder.

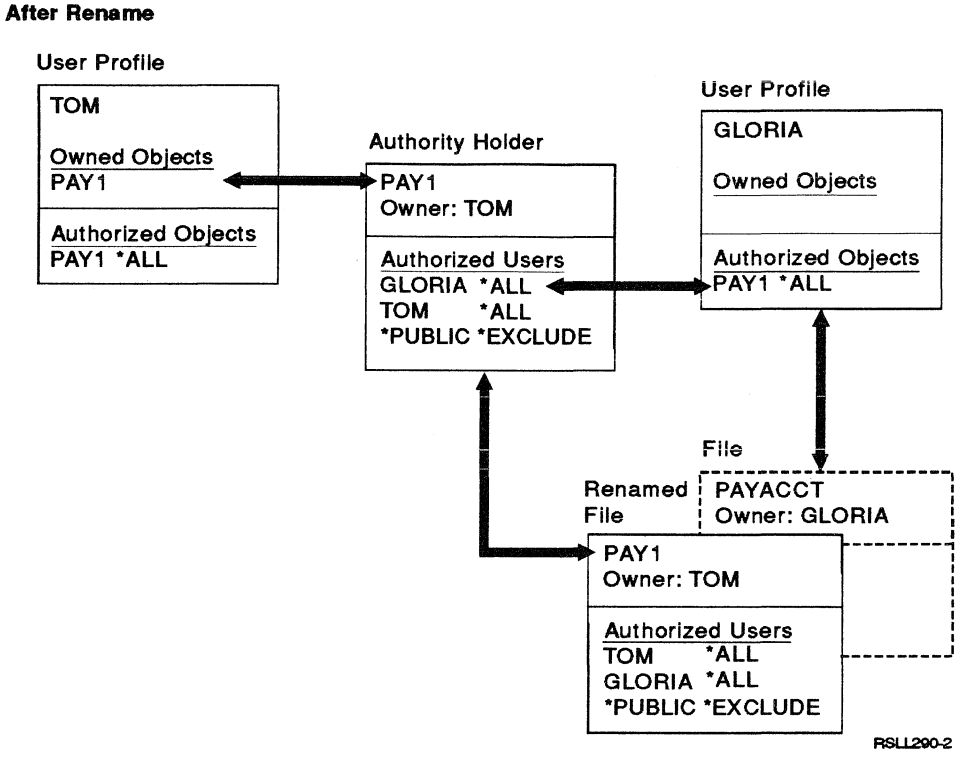


Figure 4-15. Example of an Authority Holder After Rename

When creating, deleting, or displaying authority holders, consider the following:

- When a file is created that has an authority holder by the same name, the authority specified in the authority holder is linked to the file, overriding any authority specified on the create command.
- If a file is a logical file with an authority holder, then no data authorities are used because data authorities are not valid for logical files.
- A file with an authority holder can be created only by those users with *ALL authority to the authority holder.
- If the file is created by a user that is not the owner of the authority holder, the file's ownership is changed to be the same as the owner of the authority holder.
- When a file that has an authority holder is deleted, the authority holder and its authority is kept. Therefore, when a file is re-created or restored, it is given the authority it had before the delete operation.
- When using the Display Object Authority (DSPOBJAUT) command, the authority is displayed for the object if it exists or the authority of the authority holder is displayed if the object does not exist (for example, DSPOBJAUT OBJ(PAY1) OBJTYPE(*FILE)).
- All authority holders on the system can be displayed using the Display Authority Holder (DSPAUTHLR) command.

When changing object ownership or renaming, moving or restoring files, consider the following:

- When a file is renamed and the new name is the same as an authority holder, the authority of the file is changed to be the same as that specified in the authority holder. Any authority the file had before it was renamed is no longer valid. If an authority holder does not exist for the file, the authority of the file does not change. The user renaming a file needs *ALL authority to the authority holder (if it exists) before he can rename the file.
- If an authority holder exists for a file being moved to a new library, the authority of the file is changed to be the same as the authority holder. The user moving the file to the new library must have *ALL authority to the authority holder if one exists.

  Security Consideration

  For renaming or moving a file, the owner of the file can change, but the owner will always be the same as the owner of the authority holder.

- When changing object ownership, the ownership of the authority holder (if one exists) is changed to be the same as the file.
- When an authority holder exists with the same name as the file being restored and the file is restored to the same library that is specified by the authority holder, then the file is linked to the authority holder. However, if the file is restored to a library other than the one specified by the authority holder, the restored file is not associated with an authority holder.
- When a Save System (SAVSYS) or a Save Security Data (SAVSECDTA) command is followed by a Save Library (SAVLIB(*NONSYS)) command, the authority holders and files are saved and can be restored. The authority holders can be restored using the Restore User Profile (RSTUSRPRF) command with USRPRF(*ALL) specified.

Chapter 5. Security Tips and Techniques

The purpose of this chapter is to supply some tips and techniques for security. The following is a list of some of the topics discussed for tips and techniques.

- Library list considerations
- Override commands
- Job accounting journal
- Auditing journal
- Controlling the command environment
- Controlling remote sign-on
- Controlling device descriptions
- Canceling a work station job after an inactive period
- System Request menu
- Menu security
- Submitting jobs that use adopted authority
- Job description authority
- Source files
- Controlling authority to output queues
- Using logical files
- Save and restore

In some of the discussions in this chapter, a reference is made to the library QUSRTOOL. QUSRTOOL is an optional library (may or may not be loaded) that contains some security techniques. To access the techniques mentioned in this chapter, start the source entry utility with the STRSEU command. Specify the source file QUSRTOOL/QATTINFO to display general information about QUSRTOOL. Then display the source member AAAMAP.

Library List Considerations

A **library list** is a list that indicates which libraries are to be searched and the order in which they are to be searched.

To achieve a secure system, when writing programs that use the library list function, requires special considerations. When an object is referred to without a qualified library name, the job's library list is used to find the object. If a user is able to place an object of the same name in a library before what is expected to be found on the library list, he can perform functions that break the rules of your security requirements. This can be a program, command, database file, data area, or any other object that affects security control.

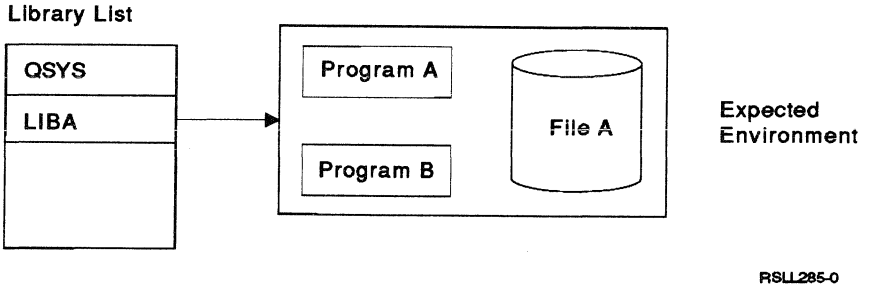
For example, USER1 is expecting to run a program named ACCTS from library LIB1, and LIB1 is on the library list. A program with the same name exists in library LIB2. If USER2 places LIB2 before LIB1 on the library list, security can be broken because ACCTS in library LIB2 is run instead of ACCTS in library LIB1.

Security Considerations

The following are controls you can use to prevent users from breaking your security requirements:

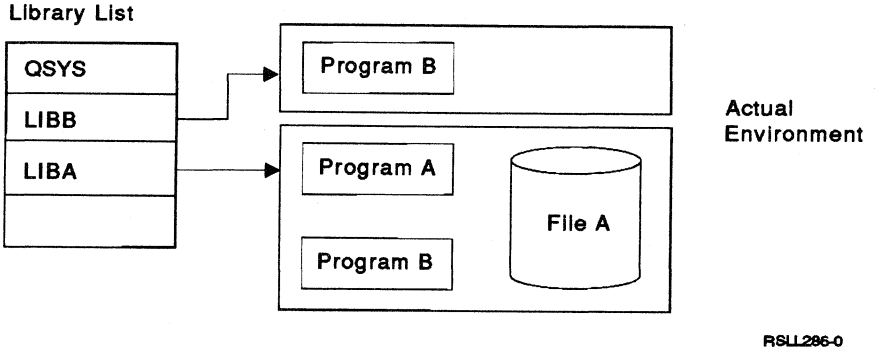
- Prevent users from using commands that allow them to change the library list by revoking authority to those commands
- Prevent users from adding objects to the libraries on the library list
- Retrieve and set the library list in an application

Of particular concern are programs that use the adopt function. For example, assume the following environment where the objects in LIBA are owned by USER1.



Program A adopts the authority of its owner (USER1) and does an unqualified call (for example, uses the library list) to call Program B. File A is authorized to USER1 and not to USER2, but File A is not used in either program.

If USER2 is allowed to place a library (for example, LIBB) in front of LIBA, on the library list and is allowed to place his own Program B in LIBB, he can gain access to the secured File A. The actual environment is shown in the following figure:



Because Program A uses the library list to find Program B, it calls the USER2 version. Because USER2 is now operating under USER1's adopted authority, USER2 can gain access to the protected file. This can occur in IBM-supplied programs, files, and commands, as well as in user-written programs; the security exposure can exist whether adopted programs are used or not.

System Portion of the Library List

Security considerations require taking into account both the system portion and the user portion of the library list.

- The system library value QSYSLIBL determines the initial setting of the system portion of the library list. This can only be changed by a user with all object (*ALLOBJ) and security administrator (*SECADM) special authorities. Only libraries that are specifically controlled should be placed on this list, and the public should not be allowed to add objects to these libraries; you should take strict control over all objects placed in these libraries. Most IBM-supplied libraries (for example, QSYS, QIDU, QRPG) are shipped so that the public cannot add objects (for example, the *ADD authority is not a *PUBLIC authority).

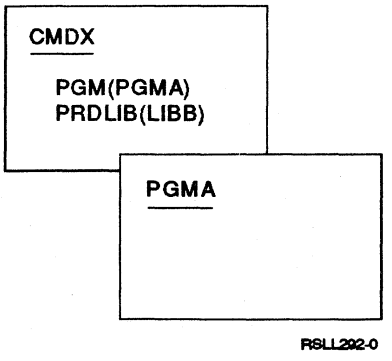
  You should ensure the *ADD authority is not a public authority for all IBM-supplied libraries except QGPL, QRJE, and QUSRSYS.

- The Change System Library List (CHGSYSLIBL) command allows the system portion of the library list to be changed within a job. You should ensure that this command is only authorized to the appropriate individuals. This command can be included in programs that adopt the all object (*ALLOBJ) special authority. For example, if a library is needed on the system portion of the library list for an entire job, it can be handled by calling a special program as part of starting the job.

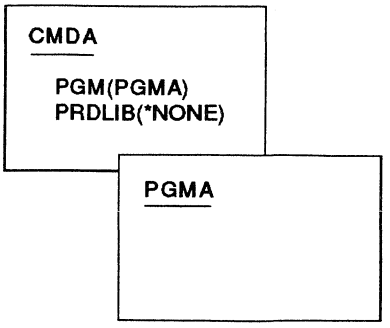
  If there is a temporary need within a job to add a library to the system portion of the library list, the same type of technique described for the user portion of the library list (see the topic "User Portion of the Library List" on page 5-4) can be used. (You must change the technique to use the Retrieve System Value (RTVSYSVAL) command of QSYSLIBL and the Change System Library List (CHGSYSLIBL) command.) The program that runs the CHGSYSLIBL command must remain in the program stack so it can return the system portion of the library list to its original version.

Product Library

A user can get the same effect as changing the system portion of the library list by creating a command that specifies a product library. For example, the following figure shows that command CMDX places product library LIBB ahead of the user portion of the library list while program PGMA is running.



To prevent this from happening, you must use a command to call program PGMA and specify PRDLIB(*NONE). PRDLIB is a parameter of the Create Command (CRTCMD) command. You should not use the default of *NOCHG. *NOCHG uses the product library from the higher call level. For example:



Current Library

A user can change the library list by using the current library (*CURLIB) function. This gives the same effect as changing the system portion of the library list.

If you are writing a program that achieves your security requirements, you should retrieve the current value and specify a new value for your processing requirements. At the end of the secure program, you should reset the value. An example of this method is discussed in the topic "User Portion of the Library List."

User Portion of the Library List

When production programs are written, they should ensure that the library list is what is expected. This is especially true of programs that use the adopt function. You must also control who can add objects to the production libraries. Several methods of achieving this follow:

- Using a protected environment such as that described for a menu security approach. If users of the protected environment cannot change the library list or add their own objects to the protected libraries, a more secure environment can be achieved.
- Using the Add Library List Entry (ADDLIBLE) command at the beginning of the program to ensure the desired objects are at the beginning of the user portion of the library list. At the end of the program, the library can be removed.

  If the library is already on the library list, but you are not sure if it is at the beginning of the list, you must remove the library and add it. At the conclusion of the function, you should replace the library to its original sequence. To replace the library to its original sequence, you may prefer the next method.
- Using the Retrieve Job Attributes (RTVJOBA) command to retrieve the current user library list and current library. Use the Change Library List (CHGLIBL) command to change to the desired library list and current library; then replace the library list to the original version at the end of the program. The following example illustrates this technique:

```
            PGM
            DCL         &USRLIBL *CHAR LEN(275)
            DCL         &CURLIB  *CHAR LEN(10)
            DCL         &ERROR *LGL
            MONMSG      MSGID(CPF0000) EXEC(GOTO SETERROR)
            RTVJOBA     USRLIBL(&USRLIBL) CURLIB(&CURLIB)
            CHGLIBL     LIBL(QGPL) CURLIB(*CRTDFT)
                        /*****************************/
                        /*                           */
                        /*   Normal processing       */
                        /*                           */
                        /*****************************/
            GOTO    ENDPGM
SETERROR:   CHGVAR      &ERROR '1'
ENDPGM:     CHGVAR &CMD ('CHGLIBL LIBL(' *CAT &USRLIBL +
                           *CAT ') CURLIB(' *CAT &CURLIB *TCAT ' )')
            CALL        QCMDEXC PARM(&CMD 500)
            IF          &ERROR SNDPGMMSG MSGID(CPF9898) +
                               MSGF(QCPFMSG) MSGTYPE(*ESCAPE) +
                                   MSGDTA('The xxxx error occurred')
            ENDPGM
```

Regardless of how the program ends (normally or abnormally), the library list is returned to the version it held when the program was called. Because the CHGLIBL command requires a list of library names, it cannot be run directly. The RTVJOBA command, therefore, retrieves the libraries used to run the CHGLIBL command through QCMDEXC.

If you exit to an uncontrolled function (for example, a user program, a menu that allows commands to be entered, or the Command Entry display) in the middle of a program, your program should replace the library list on return, to ensure adequate control.

If you use multiple libraries on the library list to locate objects, ensure that all the libraries are controlled (user has only *USE authority) so those users with public authority cannot add their own objects.

Using Override Commands

Using the override commands can also affect security because the topmost programs in the program stack control the override. You may want to specify SECURE(*YES) on the override commands to ensure the correct file is used. If users are completely controlled by menus so that they cannot use the override commands, you do not need to specify SECURE(*YES).

Job Accounting Journal

If job accounting is active, the QACGJRN journal can provide a secure object indicating which users used the system and the resources they used. For more information about job accounting, refer to the *Work Management Guide*.

Auditing Journal

If security auditing is active, the QAUDJRN journal can provide a way to monitor security related events on the system. The following events can be logged in the security auditing journal:

- Access to objects through interfaces that are not supported
- Save and restore information
- Authorization failures
- Deleted objects
- Security related functions

For more information about security auditing, see "System-Provided Security Auditing Using Journals" on page 6-10.

Controlling the Command Environment

One aspect of your system operations can include restricting certain long-running commands to be run only in a batch environment. You can do this by using the Change Command (CHGCMD) command. The following example restricts the Create RPG Program (CRTRPGPGM) command to run only in a batch environment:

```
CHGCMD CMD(CRTRPGPGM) ALLOW(*BATCH  *BPGM)
```

Controlling Sign-On for Remote Systems and PC Support

The Start Pass-Through (STRPASTHR) command allows the sign-on display at the remote system to be bypassed if the remote system is configured to allow it. The pass-through code, used to establish pass-through sessions, and the work station function will accept automatic sign-on requests and process them based on the system value QRMTSIGN set by the security officer of the remote system. The security of the remote system is the responsibility of the remote system's security officer, not that of the local system's security officer.

The QRMTSIGN system value specifies how the system handles pass-through and the work station function requests. For more information about changing the QRMTSIGN system value, see the topic "Changing the Remote Sign-On Value" on page 7-14.

The user can specify a program (qualified by a library) similar to the input and output described in the topic "Display Station Pass-Through Program" on page 5-7. This program runs at the beginning and end of each remote system pass-through and each work station function job. By this means, the security officer can tailor how automatic sign-on requests are handled and audit who has used pass-through to the remote system.

Display station pass-through is a communications function that allows a user to sign on to a system from another system and use that systems' programs and data. The *Remote Work Station Guide* has more information about the requirements for a display station pass-through program.

Display Station Pass-Through Program

```
/*****************************************************************/
/* This demonstrates the use of an exit program to control system    */
/* actions when users request remote sign-on.                        */
/*                                                                   */
/* The values returned in RTNCDE specify the system action          */
/*   0 - Do not allow remote sign-on                                 */
/*   1 - Force user to the sign-on display                           */
/*   2 - Allow remote sign-on including bypass of sign-on display    */
/*****************************************************************/
PGM  PARM(&INPUT &RTNCODE)
   /*************************************************************/
   /* Parameter declares                                        */
   /*************************************************************/
   DCL VAR(&INPUT)   TYPE(*CHAR) LEN(128)  /* Input information     */
   DCL VAR(&RTNCODE) TYPE(*CHAR) LEN(8)    /* Return code           */
   /*************************************************************/
   /* Variable declares                                         */
   /*************************************************************/
   DCL VAR(&WHENCALD) TYPE(*CHAR) LEN(1)   /* When this program is   +
                                              being called -         +
                                              '1' STRPASTHR command  +
                                              '0' ENDPASTHR command  */
   DCL VAR(&SRCPROF) TYPE(*CHAR) LEN(10)  /* Source user profile    */

   /*************************************************************/
   /*  Start of program                                         */
   /*************************************************************/
   CHGVAR &WHENCALD %SST(&INPUT 37 1) /* Determine why this program  */
                                      /* is being called             */
   IF (&WHENCALD = '0') RETURN       /* If being called for ENDPASTHR*/
                                     /* RETURN no action required.  */
   CHGVAR &SRCPROF %SST(&INPUT 17 10) /* Determine who started the   */
                                      /* request                     */
   IF (&SRCPROF = 'USER0') DO
      CHGVAR &RTNCODE '0'         /* Prevent any remote sign-on for  */
                                  /* user USER0                      */
      RETURN
   ENDDO
   IF (&SRCPROF = 'USER1') DO
      CHGVAR &RTNCODE '1'         /* Force sign-on display for       */
                                  /* user USER1                      */
      RETURN
   ENDDO
   /* All other users are allowed to use remote sign-on.   If the    */
   /* sign-on information is available, users will not see the       */
   /* sign-on display                                                */
   CHGVAR &RTNCODE '2'
ENDPGM
```

Security Considerations for Automatic Configuration of Virtual Devices

A **virtual device** is a device description that does not have hardware associated with it. It is used to form a connection between a user and a physical work-station attached to a remote system

If you allow automatic configuration of virtual devices, it will be easier for users to attempt to break in by using pass-through. Without automatic configuration, a

user attempting to break in has a limited number of attempts at each virtual device, the limit being defined by the security officer using the system value QMAXSIGN. With automatic configuration active, the actual limit is higher because the system sign-on limit is multiplied by the number of virtual devices that can be created by the automatic configuration support defined by the system value QAUTOVRT.

The system value QAUTOVRT specifies if virtual devices for normal pass-through functions (as opposed to virtual devices for work station functions) are automatically configured. This value can only be changed by the security officer or someone with all object (*ALLOBJ) and security administrator (*SECADM) special authorities.

For more information about changing this system value, see the topic "Changing the Automatic Configuration of Virtual Devices Value" on page 7-15.

Controlling Device Descriptions

A **device description** contains information about a particular device or logical unit that is attached to the system.

Devices used in a network operate like any work station device. The user must have authority to send a request. The default public authority (AUT parameter) on the Create Device Description (CRTDEVDSP) command is *LIBCRTAUT, which allows anyone to send a request. It may be desirable to exclude (*EXCLUDE) users and then give authority specifically to users that need authority.

The limit security officer (QLMTSECOFR) system value allows you to specify whether users with all object (*ALLOBJ) or service (*SERVICE) special authority can sign on to any display station. If you limit users that have *ALLOBJ or *SERVICE special authority, they must be specifically given *CHANGE authority to the specific device to use it. This also applies to anyone who is a member of a group that has *ALLOBJ or *SERVICE special authority.

When the security officer password is sent to a specific system in a call request and the security officer has not been given authority specifically to the device, the call request is rejected by the system. Anyone who has object management (*OBJMGT) and *CHANGE authority to the device can give the security officer *CHANGE authority to the device. If the security officer creates a device description, he is the owner and, therefore, is authorized to the device description. When all users (GRTOBJAUT USER(*PUBLIC) AUT(*CHANGE)) are given authority to the device, the security officer user profile is not included. This lets the security officer specify from which device security officer functions will be performed.

Canceling a Work Station Job after an Inactive Period

Two system values provide additional security to prevent users from leaving their work stations inactive. The inactive interval (QINACTITV) system value controls the time interval that a work station can remain inactive. The inactive message queue (QINACTMSGQ) determines the processing options when an inactive work station is found.

If a job is running from the work station, the job is considered active. A work station is considered inactive when there is no user interaction. Some examples of user interaction are:

- Using the Page Up and Page Down keys
- Using the Enter key
- Using the function keys
- Using the Help key

A work station is considered inactive if it is waiting at a menu or display or waiting for message input.

If an inactive message queue is specified and an inactive interval is specified, a user or program can monitor the message queue and take action as needed, such as ending the job. This allows control of specific devices at different times.

If a work station with two secondary jobs is inactive, two messages are sent to the message queue (one for each secondary job). A user or program can use the End Job (ENDJOB) command to end one or both secondary jobs. If an inactive job has one or more group jobs, a single message is sent to the message queue. Messages continue to be sent to the message queue for each interval that the job is inactive.

If the value *ENDJOB is specified for the QINACTMSGQ system value, then all jobs at the work station (secondary and group jobs) are ended by the system. A message is sent to the QSYSOPR message queue indicating that all jobs at the work station have ended.

Using a Password Approval Program

The user-written password approval program is called by the Change Password (CHGPWD) command. The user-written password approval program is controlled by the QPWDVLDPGM system value. It is recommended that the password approval program be created and placed in library QSYS in case it is necessary to recover your system from a disk failure. This way the password approval program will be loaded when you install library QSYS. Exception messages that are signaled by the program must be created with the DMPLST(*NONE) option. The following example shows how to add a message (the ADDMSGD command) to a file using this option:.

```
ADDMSGD  MSGID(BAD0001) MSGF(USERLIB/USERMSG)
   MSG('Text explaining why password not valid')
   DMPLST(*NONE)
```

If the password satisfies the system-defined password rules, the following parameters are passed to the user-written program:

1. The 10-character new password containing alphameric characters.
2. The 10-character old password containing alphameric characters.
3. A 1-character variable containing the return code that the user-written program can set to indicate a user entered a valid password (0) or a password that was not valid (not 0). The return code is initially not zero because a password that is not valid is assumed.

The following control language (CL) program is an example of a password approval program.

```
/*****************************************************************/
/* The first four lines accept the new password, old password, and */
/* location for return code and declare them.                    */
/*****************************************************************/
PGM  PARM(&NEW &OLD &RTNCD)
   DCL VAR(&NEW)   TYPE(*CHAR) LEN(10)     /* New password       */
   DCL VAR(&OLD)   TYPE(*CHAR) LEN(10)     /* Old password       */
   DCL VAR(&RTNCD) TYPE(*CHAR) LEN(1)      /* Return code        */

   /*****************************************************************/
   /* The logic to determine if the password is valid or not valid  */
   /* is placed here.                                               */
   /*****************************************************************/

   /*****************************************************************/
   /* If the password is valid, the following should be used.       */
   /*****************************************************************/
   CHGVAR VAR(&RTNCD) VALUE('0')
   /*****************************************************************/
   /* If the password is not valid, one of the following should be  */
   /* done:                                                         */
   /* The user can set the return code to nonzero, and a message    */
   /* with the return code will appear on the Change Password       */
   /* display                                                       */
   /*****************************************************************/
   CHGVAR VAR(&RTNCD) VALUE('9')
   /*****************************************************************/
   /* Or the user can signal an escape message that is already      */
   /* defined.  The message text will be displayed at               */
   /* the bottom of the Change Password display and should describe */
   /* why the password is not valid.                                */
   /*****************************************************************/
   SNDPGMMSG  MSGID(BAD0001) MSGF(USERLIB/USERMSG) +
      MSGTYPE(*ESCAPE)
   /*****************************************************************/
   /* End of program                                                */
   /*****************************************************************/
ENDPGM
```

The password is changed only if the user-written program ends with no escape message and a return code of '0'. Because the return code is initially set for passwords that are not valid (not zero), the approval program must set the return code to '0' for the password to be changed.

System Request Menu

The System Request menu provides options that are described in the *New User's Guide*.

If you do not want the user to see this menu when the System Request key is pressed, you can change the public's authority by using one of the following commands. To remove the public's authority to the panel group file, enter:

```
RVKOBJAUT OBJ(QSYS/QGMNSYSR) OBJTYPE(*PNLGRP) USER(*PUBLIC)
      AUT(*ALL)
```

To exclude the users from the panel group, type the following:

```
GRTOBJAUT OBJ(QSYS/QGMNSYSR) OBJTYPE(*PNLGRP)
      USER(*PUBLIC) AUT(*EXCLUDE)
```

Users can be prevented from using an option displayed on the menu if public authority to the corresponding command is removed. The commands associated with the options are:

Table 5-1. Options and Commands for the System Request Menu

Option	Command
1	Transfer Secondary Job (TFRSECJOB)
2	End Request (ENDRQS)
3	Display Job (DSPJOB)
4	Display Message (DSPMSG)
5	Send Message (SNDMSG)
6	Display Message (DSPMSG)
7	Display Work Station User (DSPWSUSR)
10	See note below
11	See note below
12	Display 3270 emulation options (See note below.)
80	Disconnect Job (DSCJOB)
90	Sign-Off (SIGNOFF)

Note:

1 Options 10 and 11 are only displayed if display station pass-through has been started with the Start Pass-Through (STRPASTHR) command. Option 10 is only displayed on the target system.

2 Option 12 is only displayed when 3270 emulation is active.

3 Some of the options have restrictions for the System/36 environment. See the *Concepts and Programmer's Guide for the System/36 Environment* for more information about these restrictions.

If you want to prevent users from transferring to an alternative interactive job, you can prevent them two different ways. You can enter the following command:

```
RVKOBJAUT OBJ(TFRSECJOB) OBJTYPE(*CMD) USER(*PUBLIC) AUT(*ALL)
```

or you can enter the following command to exclude users:

```
GRTOBJAUT OBJ(TFRSECJOB) OBJTYPE(*CMD) USER(*PUBLIC) AUT(*EXCLUDE)
```

If a user selects the option, a message is displayed telling the user he does not have authority to use that command.

If you revoke public authority from either the menu or one of the commands, you probably need to grant authority to specific individuals to use these functions.

If you want to prevent a user from general use of the commands from the System Request menu but still want him to be able to run a command at a specific time (such as sign-off), the security officer or any authorized user must create a CL program that adopts the profile of the security officer or that authorized user. This adopted program allows the unauthorized user to use the required command; you must also provide him with a menu option to call this program.

Menu Security

Menu security is a good approach to use when all functions on the menu control what the user can perform. For example, a menu usually prevents a user from using the Command Entry display to enter a command. However, users working with certain menu-driven programs that adopt the program owner's authority can sometimes access objects to which they are not otherwise authorized. The programs for which this is true are the following:

- The data file utility (DFU) provides read, add, update, and delete capability to files if the user has these authorities.
- Query/400 provides read capability to files if the user has authority to the file.
- SAA OfficeVision/400 allows any authorized command to be run through the run text instruction. To restrict the user from using certain commands, revoke authority from the user for those commands. OfficeVision/400 allows any authorized command to be run through the Run instruction, unless an enrollment option is taken that prevents the user from using the Run instruction.
- AS/400 Business Graphics Utility allows a user to replace a file (both the description of the file and the data in the file) if the user has object existence authority to the file.
- AS/400 BASIC provides read, add, update, and delete capability to files if the user has authority. BASIC users can also use the SYSTEM command to gain access to the Command Entry display and, therefore, to any authorized CL command. However, if the Start BASIC (STRBAS) or Start BASIC Procedure (STRBASPRC) command is run from a menu, the user can specify the CLACCESS parameter to prevent access to CL commands.
- A high-level language user can call QCMDEXC to run any authorized command or can call QCMD or QCL to gain access to the Command Entry display.

Security Consideration

The limited capability (LMTCPB) parameter in the user profile is designed to prevent a user from entering commands on the command line of an IBM-supplied menu. It is recommended that LMTCPB(*YES) be specified in user profiles that use menu security. The CHKLMTCPB command in library QUSRTOOL assists in checking and controlling all *USER type profiles.

You can also control what menu options a user can select by revoking the user authority to the command that is run when the option is selected. To find out what command is associated with a system menu option, you can display the online help information for that menu option. The command name is usually enclosed in parentheses at the top of the online information.

The following figure shows an approach to menu security that allows controlled access to the functions provided by AS/400 Query, date file utility, OfficeVision/400, and BASIC.

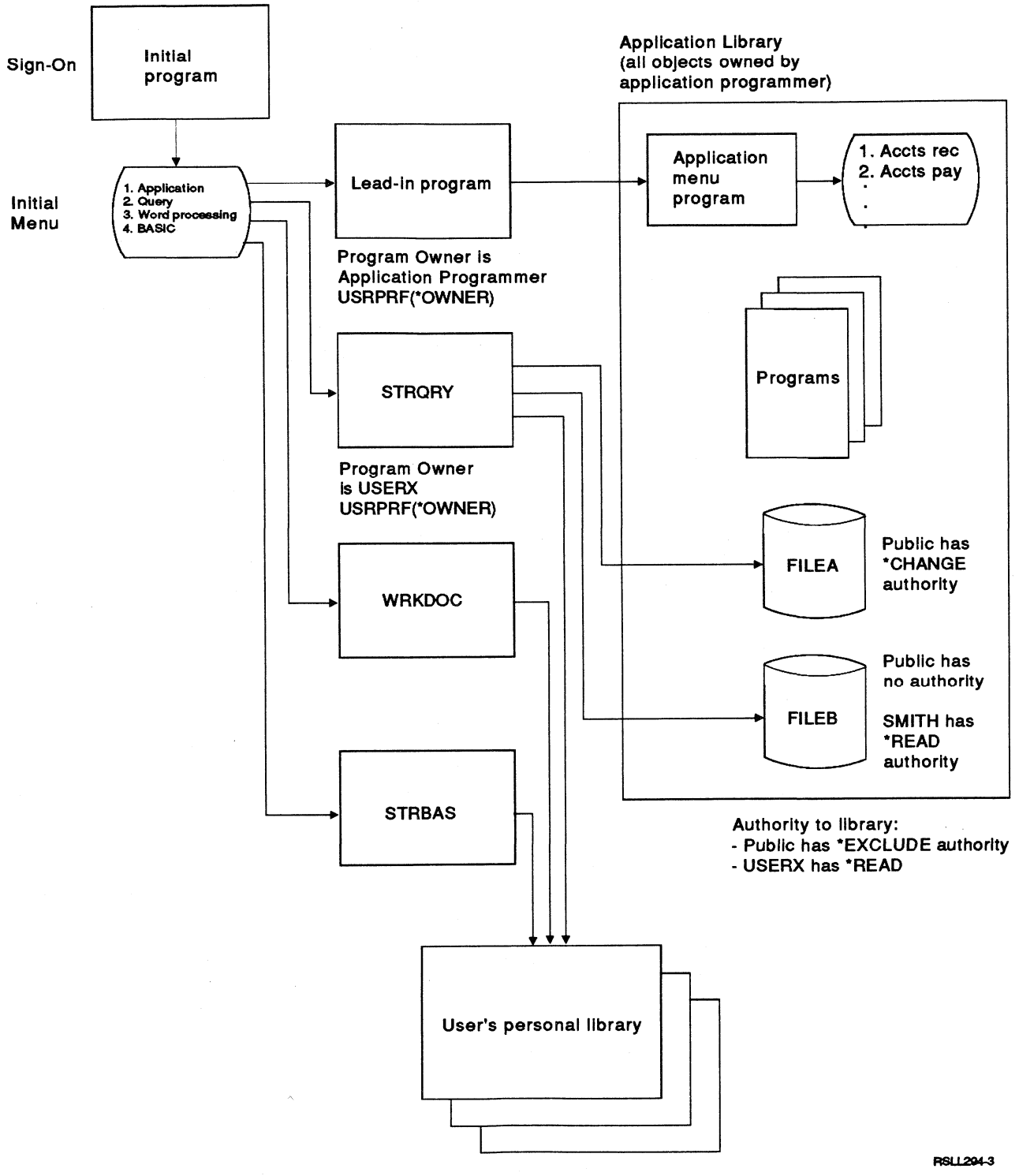


Figure 5-1. Example of Menu Security

A user with a user profile named application programmer (APPPGMR) owns the application program objects. When any authorized user signs on, the initial program displays a menu that allows him to enter into application program functions or to access user functions provided by AS/400 Query, word processing, or BASIC. The initial program does not adopt the owner's profile, so the user is operating under his own profile authorizations at this point.

If the application option is chosen from the initial menu, a lead-in program is called. The lead-in program is owned by APPPGMR and adopts his authority. It sets the library list (or uses the Add Library List Entry (ADDLIBLE) command) to allow access to the application program functions and calls the program that displays the first menu. If application programs are written specifically for the user, the lead-in program can be standard for all users by having the name of the application program passed as a parameter and then called.

The first application program exists in the application program library, and all application program objects can be accessed because the lead-in program adopts the APPPGMR authority and remains in the program stack. When the application program returns, the lead-in program resets the library list (or uses the Remove Library List Entry (RMVLIBLE) command) and returns to the initial program. This removes the APPPGMR adopted authority, and the user returns to his own authority. The public authority for the lead-in program should be specified as exclude (*EXCLUDE), and only users who will be using this program should be granted authority to use the program.

A user who wants to do a query selects that option from the initial menu. A program that is owned by another user profile is called; this program adopts that owner's profile. (In the example, Figure 5-1 on page 5-13, the user profile is called the USERX profile and is authorized to access the application program library but has no authority to any other function.) The program runs the Start Query (STRQRY) command, which allows the user to design and run a query. The user has access to the application program library and can query any publicly authorized file or any file to which he has specific authority. In the figure, FILEA can be queried because it is publicly authorized. FILEB can only be queried if the user is SMITH. Any objects created from query should be directed to the user's personal library. This allows him to control who has access to these objects. The program owned by USERX exists in a library that is available to all users. This program should be privately authorized to users who are allowed to use query on the files in the application program library.

Notice that user SMITH can also perform the application function against FILEB. The application program can update this file even though SMITH has only *READ authority because of the adopt function performed by the lead-in program.

If the user selects word processing or BASIC from the initial menu, the application program directly calls the correct command (no intervening program is needed to call BASIC or word processing). This means that the functions are running under the user's profile, and he has no authority to access the application program libraries. He can access only objects that are in his personal library.

If the word processing user needs access to the database files in the application program library, the same technique can be used as is shown for the query function. It might be desirable to split the application library into a library for database files and a library for other objects. The USERX profile should only be given access to the library containing database files. This prevents the word processing user from running a command using the production objects other than files. Then only the database objects need to be controlled with specific authorizations. For example, the public or specific users can be given the authority to read the file, but not to update it or delete records.

There are cases in which the application program is developed by programmers other than the owner of the application library. These development program-

mers can be prevented from accessing the application library because it is private. If they need the ability to copy data for test purposes, an adopt type of menu program owned by the APPPGMR profile can be used to limit their capability. The same approach as shown for the application users can also be used if the programmers are allowed to use query on application database files.

Note: For an example, see ACCSECLIB in library QUSRTOOL.

The system operator may need to submit jobs or perform interactive actions on the application program library. A menu program owned by APPPGMR using the adopt function can be used to ensure that he only runs specific functions using the application program objects. To adequately use a menu approach to security, you should refer to the topic "Library List Considerations" on page 5-1.

Submitting Jobs That Adopt Authority

The previous topics describe application programs that are run interactively. This topic discusses how end users who submit batch application programs that run under the owner's authority can adopt authority to operate on objects in the application program library.

An approach is to use a special routing entry and a program that is designed for the purpose of batch jobs. For example:

```
ADDRTGE SBSD(QBATCH) SEQNBR(600) CMPVAL(ADOPT) PGM(ADOPTC)
```

The program can be written as:

```
PGM
CALL QCMD
ENDPGM
```

The program ADOPTC must be created with USRPRF(*OWNER).

The system calls ADOPTC if the correct routing data is specified. The call to QCMD allows a submitted job operation where you specify a command to run with the command (CMD) or the request data (RQSDTA) parameter. For example:

```
SMBJOB JOB(XXX) RTGDTA(ADOPTC) RQSDTA('CALL XXX')
```

Your interactive application programs need to specify only the CMD parameter to use the ADOPTC program. With some simple additions to the subsystem, you can achieve the function of adopting in batch without a significant effect to the application programs.

For example, if you have separate profiles that you want to adopt authority at different times, you can add a routing entry for each unique situation. For example, assume that you have a routing step that adopts the MILLERK user profile and another routing step that adopts the SMITHP profile. Each routing entry would need to specify a unique program that uses the owner's authority (USRPRF(*OWNER)). For unique situations where programs use more than one user profile that adopts authority, you can write a series of programs such as:

```
ADOPTC  Pgm  Owned by SMITHP with USRPRF(*OWNER)
   CALL ADOPTC2
ADOPTC2 Pgm  Owned by MILLERK with USRPRF(*OWNER)
   CALL QCMD
```

When the user submits a job with RTGDTA(ADOPTC), the ADOPTC program adopts SMITHP, and the ADOPTC2 program adopts MILLERK. These routing steps combine all of the authorities of SMITHP, MILLERK, and the user of the job while the batch job is running.

Note: You should give authority for program ADOPTC only to those users who need authority. This can exclude the application programmers.

Job Description Authority

To create a job description with a specific user profile name requires the user of the Create Job Description (CRTJOBD) command or the Change Job Description (CHGJOBD) command to have use (*USE) authority to the user profile.

When the system security level is 40, *USE authority is also required for the user profile specified in the job description by the user submitting batch jobs. At security level 30, attempts to submit a batch job will record a journal entry in the auditing journal if the user does not have *USE authority to the user profile specified in the job description (JOBD). At security level 30, the user does need *USE authority to the user profile specified in the job description.

On a regular basis, you may want to:

- Review all work station entries and the job descriptions being used for interactive jobs to ensure that the job descriptions that have specific user profile names specified meet security requirements.
- Review all job descriptions that have a specific user profile name in the USER parameter to determine if authority to these job descriptions meet security requirements. If a job description has a specific user profile name and has public authority of *USE, then anyone can use it. If you have a security-sensitive job description, specify public authority of *EXCLUDE and then give authority to those users who need to use the job description.

  Note: For an example, see the CHKJOBDUSR command in library QUSRTOOL.

- Review the authority to user profiles to ensure that public authority for the user profiles meet security requirements.

Note: Frequently, a specific user profile name in a job description is required to let users submit work for a specific user profile. For example, the QBATCH job description is shipped with USER(QPGMR) to allow this. This job description is created with the public authority of *CHANGE. This means that any user on the system who has authority to the Submit Job (SMBJOB) command or the start reader commands can submit work under the programmer (QPGMR) user profile. You may want to change the public authority to *EXCLUDE, depending on your security needs. At security level 40, *USE authority is required to the user profile specified in the job description in order to start a job using the job description.

Controlling Authority to Output Queues

This topic explains the security for output queues. This topic does not explain how to send output to a specific printer, but it does specify which users are allowed to perform operations on output queues and on spooled files.

There are several different levels of authority to an output queue. The level of authority depends on the operations the user needs to perform:

- Working with all output queues
- Working with the items on the output queue
- Displaying the content of the spooled files on the output queue
- Working with the spooled files (change, delete, and so on)

The level of authority for an output queue and for the spooled files on an output queue is determined by parameters in both the user profile and in the output queue itself.

Table 5-2 on page 5-18 summarizes the parameters shown on the following display that affect output queue security.

```
                      Create Output Queue (CRTOUTQ)

Type choice, press Enter.
Output queue . . . . . . . . . . OUTQ
  Library  . . . . . . . . . .                  *CURLIB
Order of files on queue  . . . . SEQ
Text 'description' . . . . . . . TEXT           *BLANK


                      Additional Parameters
Display any file . . . . . . . . DSPDTA         *NO
Job separators . . . . . . . . . JOBSEP         8
Operator controlled  . . . . . . OPRCTL         *YES
Authority to check . . . . . . . AUTCHK         *OWNER
Authority  . . . . . . . . . . . AUT            *USE



F3=EXIT   F4=Prompt   F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

Figure 5-2. Create Output Queue Display

Table 5-2. User Profile and Output Queue Parameters Affecting Security

Command	Parameter	Value	Meaning
CRTOUTQ Create Output Queue[1]	DSPDTA Display any file	*NO	Users authorized to use the output queue can display, copy, or send the output data of the spooled files they have created, unless they have some other authority that overrides this[3].
		*YES	Any user having authority to read the output queue can display, copy, or send the data of any file on the queue.
	OPRTCL Operator controlled	*YES	A user with job control special authority in the user profile can control the output queue and make changes to the files on the output queue.
		*NO	The output queue and its entries cannot be controlled or changed by users with job control special authority, unless they also have some other authority that overrides this[3].
	AUTCHK Authority to check[2]	*OWNER	Specifies that only the owner of the output queue can control all the output on the queue.
		*DTAAUT	Specifies that any user with read, add, and delete authority to the output queue can control all output files on the queue.
	AUT Authority	*USE	*USE authority allows the user to perform basic operations on all the output queues, such as send output to the output queues.
		*CHANGE	Allows the user to change the output queue description. To control files created by other users, create the output queue with the value *DTAAUT specified for the *Authority to check* prompt (AUTCHK parameter).
		*ALL	Allows the user to perform all operations on the output queue except those limited to the owner.
		*EXCLUDE	Prevents the user from accessing the object, unless the user has some special authority.
CRTUSRPRF Create User Profile	SPCAUT	*JOBCTL	Allows the user with *JOBCTL special authority to change, display, hold, release, cancel and clear all jobs that are on an output queue (as well as jobs running on a job queue). This occurs if the output queue is specified as operator controlled (OPRCTL(*YES)).
		*SPLCTL	Allows the user with *SPLCTL special authority to perform all operations on the output queue. Therefore, only the security officer should have *SPLCTL special authority.

Notes:

1 See the Create Output Queue (CRTOUTQ) command.

2 The parameter AUTCHK (Authority to check) specifies the type of authorities to the output queue that allows the user to control all the files on the queue. Users with some other authority may also be able to control the output files (see Table 5-2 on page 5-18).

3 See Table 5-3 on page 5-19.

The following tables (Table 5-3, Table 5-4, and Table 5-5 on page 5-20) show the authority a user has, depending on different combinations of the parameters in the user profile and in the definition of the output queue.

Table 5-3. Combination of User Profile and OUTQ Parameters - Example 1

User Profile SPCAUT	**OUTQ OPRCTL**	**Capabilities**
*JOBCTL	*YES	User allowed to: Look at the output queue Look at the spooled files in the output queue Add new spooled files to the output queue Delete and change spooled file entries
*SPLCTL	N/A	

Table 5-4. Combination of User Profile and OUTQ Parameters - Example 2

User Profile SPCAUT	**OUTQ OPRCTL**	**Capabilities**
*JOBCTL	*NO	See Example 3.
Other than *JOBCTL or *SPLCTL	*YES N/A	

Table 5-5. Combination of User Profile and OUTQ Parameters - Example 3			
OUTA PARAMETERS			
OUTQ Object Authority	**DSPDTA**	**AUTCHK**	**Capabilities**
*CHANGE	N/A	*DTAAUT	User allowed to: Look at output queue Look at spooled file entries Add new spooled files Change and delete entries
*CHANGE	*YES	*OWNER	User allowed to: Look at output queue Look at spooled file entries Add new spooled files User NOT allowed to: Change and delete entries
*CHANGE	*NO	*OWNER	User allowed to: Look at output queue Add new spooled files User NOT allowed to: Look at spooled file entries Change and delete entries
*USE	*YES	N/A	User allowed to: Look at output queue Look at spooled file entries Add new spooled files User NOT allowed to: Change and delete entries
*USE	*NO	N/A	User allowed to: Look at output queue Add new spooled files User NOT allowed to: Look at spooled file entries Change and delete entries
*EXCLUDE	N/A	N/A	User NOT allowed to: Look at output queue Look at spooled file entries Add new spooled files Change and delete entries

Source Files

Source files are treated like database files on the system. The default is that:

- Only the owner can add or remove a member
- Any user can update the data

Because the source entry utility (SEU) has functions that allow the user to add or remove members, SEU checks to ensure that the user has object management (*OBJMGT) authority to the file to use SEU for adding or updating a member. A specific authority is required for this function (it is not the default of the Create Source Physical File (CRTSRCPF) command). However, the default public authority to the source file is the same as any other data file. Thus, a user with change authority can open, read, and update data through a program other than

SEU. To control this, you should give individual authorities for the file and revoke public authority if necessary.

Security Consideration

Because source files are important to the integrity of any system, you may want to place them in separate libraries and allow only certain user profiles to update them.

For example, you can allow any user to read the source, but allow only certain user profiles to update the source. You can do this by revoking the public authorities of update, add, and delete.

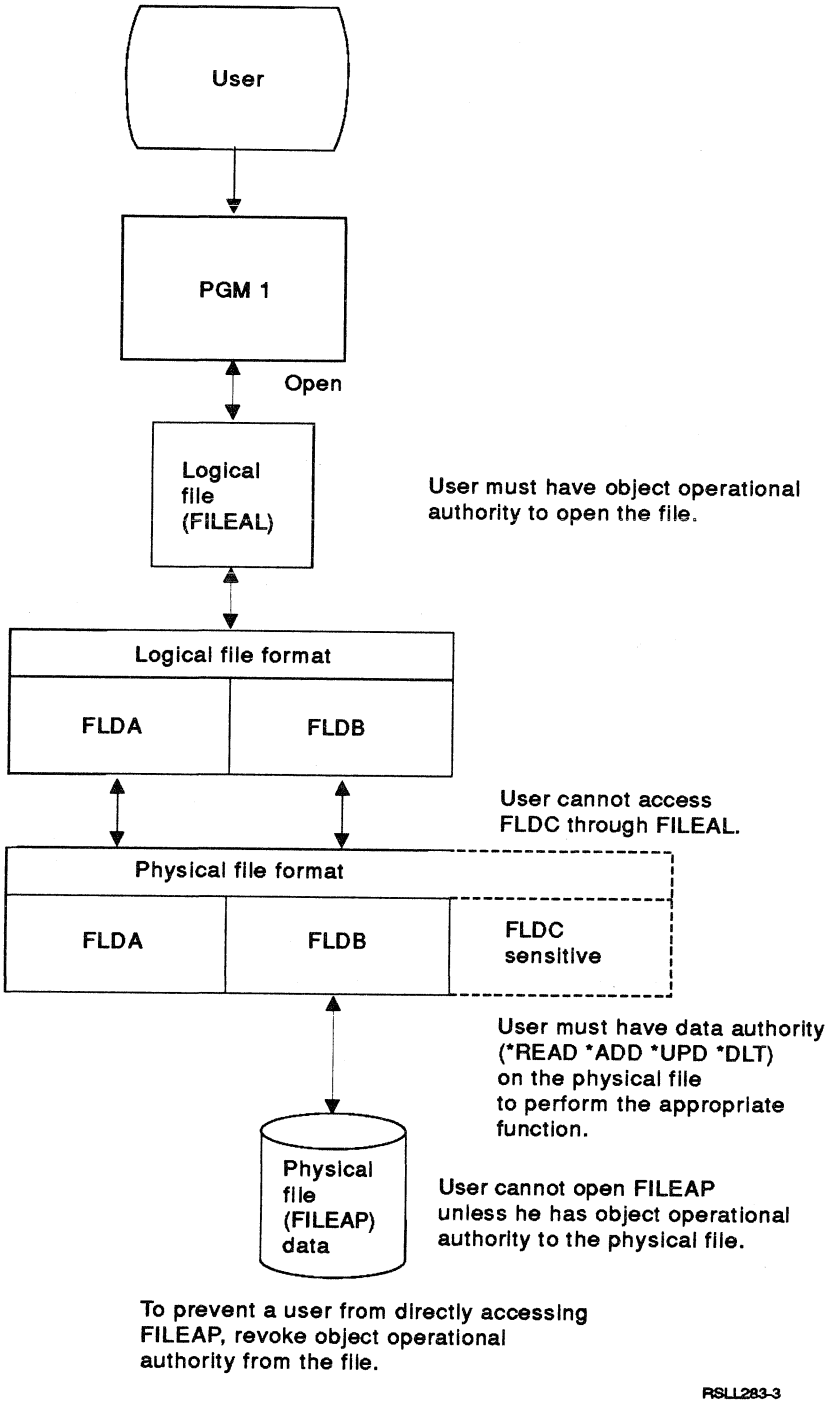
For more information on the authority needed to perform SEU operations on a file member, see the *SEU User's Guide and Reference*.

Using Logical Files

Database physical files may contain data that is sensitive and should not be changed or displayed by certain users. The system supports security at a file level, but you can design a logical file that does not contain the sensitive fields.

A logical file can be used to specify a subset of *records* that a user can access (by using select and omit logic). Therefore, specific users can be prevented from accessing certain record types. A logical file can be used to specify a subset of *fields* in a record that a user can access. Therefore, specific users can be prevented from accessing certain fields in a record. The following figure shows how you can use logical files to protect sensitive fields from an unauthorized user.

Note: Data authority is not allowed for a logical file. All data authority is checked from the physical file. Only *OBJOPR, *OBJMGT and *OBJEXIST authorities can be granted to a logical file.



Save and Restore Operations

The save and restore operations allow an object created on one AS/400 system to be restored on another AS/400 system. The following table shows how the different types of security data are saved and restored.

Table 5-6. Objects Saved or Restored by Commands					
What is saved or restored?	Save and Restore Commands Used				
	SAVSECDTA SAVSYS	SAVLIB SAVOBJ SAVDLO	RSTUSRPRF	RSTLIB RSTOBJ RSTDLO	RSTAUT
User profiles	X		X		
Authorization lists	X		X		
Authority holders	X		X		
Private authorities	X				X
Public authorities		X		X	
Link with the authorization list and authority holders		X		X	

Saving and Restoring Objects and Saving the Security Information

Consider the following when saving and restoring user profiles:

- The Save System (SAVSYS) or the Save Security Data (SAVSECDTA) command saves all user profiles (and any private authorities they have), authorization lists, and authority holders on the system.
- The Restore User Profile (RSTUSRPRF USRPRF(*ALL)) command restores all the user profiles, authorization lists, and authority holders on the system.
- The Restore Authority (RSTAUT) command restores private authorities for the user profiles after the user profiles and objects are restored.

Consider the following when saving and restoring objects:

- The Save Library (SAVLIB), Save Object (SAVOBJ), Save Changed Object (SAVCHGOBJ), and the Save Document Library Objects (SAVDLO) commands save the name of the owner, the public authority, the authorization lists, and the authority holders that secure the object.
- The Restore Library (RSTLIB), Restore Object (RSTOBJ), and Restore Document Library Objects (RSTDLO) commands restore the objects, including the public authority and the link that the object has with an authorization list or authority holder.

  If an object exists on the system, the public authority from the save media is not used. If an object does not exist, the public authority is restored from the save media and not set using the CRTAUT value of the library where the object is being restored into.

  During a save operation, if an object is secured by an authorization list, the name of the authorization list is saved with the object.

  During a restore operation, the following rules apply to restoring an object that was secured by an authorization list when it was saved:

  - If the object is being restored on the same system from which it was saved, the object is linked to the authorization list again.
  - If the object is being restored on a different system, then *ALL must be specified for the ALWOBJDIF parameter on the Restore command in order for the object to be linked to the authorization list. If the ALWOBJDIF(*ALL) is ***not*** specified, then the object is not be linked to the

authorization list and the public authority of the object is changed to *EXCLUDE.

- If the object already exists on the system, it must have the same authorization list as the object on the media or online save file. If not, the object is not restored.
- The authorization list must exist on the system. If not, the object is restored without being linked to an authorization list and the public authority is changed to *EXCLUDE.

Saving the System Security Information

Security information can be saved to tape or to a save file without doing a complete save system (SAVSYS command) and without using a restricted system. The Save Security Data (SAVSECDTA command) saves the same security information as the SAVSYS command. A user must have save system (*SAVSYS) special authority to use the SAVSECDTA command. For more information about the SAVSYS command, see the *Backup and Recovery Guide*.

The following information is saved using the SAVSECDTA command.

- User profiles
- Authorization lists
- Authority holders
- Office distribution objects

The security information can also be saved to a specified save file. The data in the save file can be saved to tape using the Save Save File Data (SAVSAVFDTA) command.

A restricted state is required when restoring the information that was saved to tape with the SAVSECDTA command. For more information about placing the system in a restricted state and restoring the security data, see the *Backup and Recovery Guide*.

Restoring Programs That Adopt the Owner's Authority

Certain restrictions minimize possible security exposures in restoring programs that adopt the owner's authority. Private and public authorities to a restored program are maintained only when it is the owner or the security officer who is restoring the program. The system determines this by verifying the following:

1. The user profile of the job doing the restore operation
2. Whether the user profile doing the restore operation has a group profile that is the owner or the security officer (*ALLOBJ and *SECADM special authorities)
3. Programs that adopt the owner's user profile during the restore operation

Consider the following when restoring a program that adopts the owner's authority:

- If the program is restored by someone other than the owner or the security officer, all public and private authorities are revoked, and the public authority is changed to *EXCLUDE.
- If the owner does not exist on the system when the program that adopts the owner's authority is restored, ownership is given to the QDFTOWN user

profile. Public authority is changed to *EXCLUDE authority and the authorization list is removed.

Restoring User Profiles

Consider the following when restoring user profiles:

- When restoring an individual user profile, the following considerations apply to the user profile being restored:

  If the user profile does not exist on the system at the time of the restore operation, but exists on the save media when the restore operation is performed, then the group profile information, password, and document password are all set to *NONE. All other user profile values are restored from the media without being changed by the system.

  If a user profile existed on the system at the time of the save operation, and exists when the restore operation is performed, then the group profile information, password, and document password are not changed by the system. All other user profile values are restored from the media without being changed.

- When restoring user profiles with *ALLOBJ special authority to a level 30 system or above, only the system (QSYS) user profile, the security officer (QSECOFR) user profile, and the install profiles (QLPAUTO and QLPINSTALL) are restored with all object (*ALLOBJ) special authority. All other user profiles have the *ALLOBJ special authority removed for security reasons.

Security Consideration

You may want to control the media (diskettes or tapes) in a safe or a fireproof vault.

See the *Backup and Recovery Guide* for a further discussion of save and restore security considerations.

Recovering from a Damaged Authorization List

When an object is secured by an authorization list and the authorization list becomes damaged, access to the object is limited to users that have all object (*ALLOBJ) special authority.

To recover from a damaged authorization list, two steps are required:

1. Recover users and their authorities on the authorization list.
2. Recover the association of the authorization list with the objects.

The steps listed above can be done by a user with *ALLOBJ special authority.

Recovering the Authorization List

If the users on the authorization list are known, simply delete the authorization list, create the authorization list again, and then add users to it.

If it is not possible to create the authorization list again because you do not know who was on the authorization list, the authorization list can be restored and the users restored to the authorization list using your last SAVSYS or SAVSECDTA tapes. To restore the authorization list, do the following:

1. Delete the damaged authorization list.

```
DLTAUTL AUTL(authorization-list-name)
```

2. Restore the authorization list by restoring the user profiles.

   ```
   RSTUSRPRF USRPRF(*ALL)
   ```

3. Add users to the list by restoring authority.

   ```
   RSTAUT
   ```

Consideration

The above procedure restores any authorities and user profiles that have been deleted since the last save operation

Recovering the Association of Objects to the Authorization List

When the damaged authorization list is deleted, the objects secured by the authorization list need to be added to the new authorization list. Do the following:

1. Find the objects that were associated with the damaged authorization list using the Reclaim Storage (RCLSTG) command.

   ```
   RCLSTG
   ```

   Reclaim storage assigns the objects that were associated with the authorization list to authorization list QRCLAUTL.

2. Use the Display Authorization List Objects (DSPAUTLOBJ) command to get the names of the objects associated with the damaged authorization list. If a large number of objects are found, use the OUTFILE parameter to create the list in a database file.

   ```
   DSPAUTLOBJ AUTL(QRCLAUTL)
   ```

3. Use the Grant Object Authority (GRTOBJAUT) command to grant the authorization list authority to each object in the list.

   ```
   GRTOBJAUT OBJ(library-name/object-name) AUTL(authorization-list-name)
   ```

Chapter 6. Auditing Security for the AS/400 System

Security auditing, as used here, refers to two schedules:

- Monitoring security daily
- Monitoring security periodically

The day-to-day monitoring should be part of security administration. The security officer will probably perform these tasks every day.

Periodic security audits may be performed by internal or external auditors. The frequency of audits depends on the size and security needs of an organization. The purpose of this chapter is to discuss why these activities are needed rather than to give guidelines for their frequency.

Security monitoring and auditing involves using commands on the AS/400 system and accessing log and journal information on the system.

The monitoring and auditing tasks suggested in this chapter require a user profile with *ALLOBJ and *SECADM special authority.

Monitoring Security Daily

Security, when established at the desired level, tends to change over time. The dynamics of the computer and the complexity of the environment are the reasons for this. Some typical examples are:

- New objects created by system users
- New users admitted to the system
- Change of object ownership (authorization not adjusted)
- Change of responsibilities (user group changed)
- Temporary authority (not timely revoked)
- New products installed

It is necessary that the most important (primary) security controls be monitored regularly in the following two categories:

- Analyzing primary security events
- Verifying the status of primary security controls

Monitoring the Status

The primary day-to-day activity to maintain good security is monitoring the status of primary security controls. Analyzing recently recorded security events adds to this activity.

Primary security controls to be checked are:

- General controls and options at the system level
- User profiles and authorities
- Object descriptions and authorizations

The primary security controls at the system level are:

- The system security options
- The keylock switch position

Verifying System Security Options

The primary option is the security level (QSECURITY), which must be set to 30 or above for any secure environment.

The other option is the sign-on limit (QMAXSIGN); it is recommended that this value not exceed 5. Using a menu approach, these values can be verified through the following procedure.

On the AS/400 Main Menu:

1. Select option 7 (Define or change the system).
2. Select option 8 (Work with system values).
3. Select the option to display the following system values:
   - QSECURITY to display the security level
   - QMAXSIGN to display the sign-on limit

Alternatively, you can type the following commands on the command line to display the system value options:

- DSPSYSVAL QSECURITY
- DSPSYSVAL QMAXSIGN

Verifying Keylock Switch Setting

Set the security keylock switch to the Secure or the Auto position, and remove the key from the AS/400 system. Keep the key under tight physical and procedural controls. Verify the keylock switch setting by visually inspecting it.

Monitoring Critical User Profiles

Check critical user profiles regularly; they are user profiles that have special authorities and IBM-supplied user profiles for which the default passwords are provided.

User Profiles with Special Authorities

Look at all user profiles with special authorities, such as *ALLOBJ authority. Compare these user profiles with the list of authorized users. Include other values in your analysis, such as PASSWORD(*NONE).

The Display Authorized Users (DSPAUTUSR) command can print the following information for all user profiles:

- User profile name
- Group profile name
- Date password was last changed
- An indication if the password is *NONE
- Description text

To print a list of authorized users, type:

```
DSPAUTUSR OUTPUT(*PRINT)
```

To print other user profile information, type:

```
DSPUSRPRF USRPRF(user-profile-name) TYPE(*ALL) OUTPUT(*PRINT)
```

Along with basic profile information, the following is printed: all commands, devices, and objects that the user has specific authority for, objects the user owns, and group members (if the profile is a group profile).

IBM-Supplied User Profiles

Check IBM-supplied user profiles in the following ways:

- For user profiles designed as object owners only, verify that they cannot be used to sign on to the system (password should be *NONE).
- For user profiles shipped with default passwords, verify that these passwords cannot be used to sign on to the system.

Change the default passwords immediately after installing the system. In addition, they should be changed periodically (in case they become known or they are reset to the defaults). The IBM-supplied user profiles at security level 30 or above have the characteristics shown in Table 6-1.

Table 6-1. IBM-Supplied User Profiles. The AS/400 system is shipped with these user profiles built into the system. Other than the passwords, do not change them.

User Profile	Password	*ALLOBJ	*SAVSYS	*JOBCTL	*SECADM	*SPLCTL	*SERVICE	Class	Group
QDBSHR	*NONE							*USER	*NONE
QDFTOWN	*NONE							*USER	*NONE
QDOC	*NONE							*USER	*NONE
QFNC	*NONE							*USER	*NONE
QGATE	*NONE							*USER	*NONE
QLPAUTO	*NONE	Yes	Yes	Yes	Yes			*SYSOPR	*NONE
QLPINSTALL	*NONE	Yes	Yes	Yes	Yes			*SYSOPR	*NONE
QPGMR	QPGMR		Yes	Yes				*PGMR	*NONE
QRJE	*NONE		Yes	Yes				*PGMR	*NONE
QSECOFR	QSECOFR	Yes	Yes	Yes	Yes	Yes	Yes	*SECOFR	*NONE
QSNADS	*NONE							*USER	*NONE
QSPL	*NONE							*USER	*NONE
QSPLJOB	*NONE							*USER	*NONE
QSRV	QSRV		Yes	Yes			Yes	*PGMR	*NONE
QSRVBAS	QSRVBAS			Yes			Yes	*PGMR	*NONE
QSYS	*NONE	Yes	Yes	Yes	Yes	Yes	Yes	*SECOFR	*NONE
QSYSOPR	QSYSOPR		Yes	Yes				*SYSOPR	*NONE
QTSTRQS	*NONE							*USER	*NONE
QUSER	QUSER							*USER	*NONE

Verify the IBM-supplied user profiles to ensure that the passwords for the following user profiles have been changed:

- QSECOFR
- QSYSOPR
- QPGMR
- QUSER
- QSRV
- QSRVBAS

Monitoring Critical Objects

For critical objects, check the public and specific authority. Some of the critical system objects are:

```
Object    Type   User       Authority
_______   ____   ________   _________
QSYS      *LIB   *PUBLIC =  *USE
QUSRSYS   *LIB   *PUBLIC =  *USE
QHLPSYS   *LIB   *PUBLIC =  *USE
```

Critical site objects are production libraries containing programs, source programs, and files used in application programs with high protection requirements (for confidentiality or accuracy reasons). They should be added to the list above.

Monitoring Journals and History Log

The history log and journal files contain security-related events and other information that must be monitored. This information must be analyzed and described in security reports for management review. The following priorities are suggested:

- Analyze reported changes to security requirements and rules
- Analyze access granted to critical objects
- Analyze attempted misuse

Use the Display Log (DSPLOG) command to display the history log (QHST) and look for messages in the range of CPF2200 (these messages are related to security). If the QAUDLVL system value has been set to *AUTFAIL or *PGMFAIL, you can check the journal entry type AF in the auditing journal QAUDJRN using the Display Journal (DSPJRN) command.

Analyzing Authority for Critical Objects

Security requirements and rules tend to become less effective over time. A periodic review of all rules (for an application program) is the best way to correct this situation. While this approach may be acceptable for the majority of objects at a site, some rules may not be acceptable for some highly critical objects. For these, you should frequently monitor and verify where the system is granting access.

You can print access authority for a specific object with the following command:

```
DSPOBJAUT OBJ(library/object) OBJTYPE(type) OUTPUT(*PRINT)
```

For program SINGLE in library COOPERS, you would type:

```
DSPOBJAUT OBJ(COOPERS/SINGLE) OBJTYPE(*PGM) OUTPUT(*PRINT)
```

For a computer printout of users on an authorization list, use the following command:

```
DSPAUTL AUTL(XYZ) OUTPUT(*PRINT)
```

where XYZ is the name of the authorization list.

Analyzing Changes to Security

Display or print a list of users using the Display Authorized User (DSPAUTUSR) command. Review the list and delete users who are no longer valid users.

Analyzing Attempted Misuse

The AS/400 system records such events as using incorrect passwords, attempting to access an object with insufficient authority, and so on. These events are recorded in the history (QHST) log and the auditing journal (QAUDJRN) (if it exists). Examples of specific commands for a list of QHST messages are shown in the topic "Monitoring Security Using History Log Commands" on page 6-7. For more information about journal QAUDJRN, see the topic "System-Provided Security Auditing Using Journals" on page 6-10.

Using Journals

You can record all changes and accesses to physical files by using journals. While the use of journals is related more to application design than overall *system* security, the auditor needs to understand their use at a given site. A journal can include:

- Identification of the job and user, and the time of access
- Before- and after-images of all file changes
- Records of when the file was opened, closed, and saved

A journal entry cannot be altered by any user, even the security officer. A complete journal can be deleted, but this is easily detected.

If you are journaling and want to print all information about a particular file, type the following:

```
DSPJRN JRN(library/journal) FILE(library/file) OUTPUT(*PRINT)
```

If journal JOURNAL in library COOPERS is used to record information about file USRINLC (also in library COOPERS), the command would be:

```
DSPJRN JRN(COOPERS/JOURNAL) FILE(COOPERS/USRINLC) OUTPUT(*PRINT)
```

Monitoring Security Periodically

You can monitor security at various levels of detail. A diagnostic review might be limited to answering general questionnaires. More detailed reviews would analyze the system status, verify the security requirements with actual use, or include a statistical analysis of the security requirements and, if appropriate, program code review.

Analyzing User and Group Authority

You need to analyze the system users, and their organization in groups.

The following items relate to the security policies regarding individual and group users.

1. The passwords have been changed for the IBM-supplied user profiles.
2. The number of users with special authorities is reasonably small.
3. Multiple sign-on with the same user ID are prohibited.
4. Password changes are required and enforced at appropriate intervals (for example, 30 days).
5. Group profiles have PASSWORD(*NONE) specified.
6. An organization chart exists for all system users.
7. The administration of user profiles is adequately organized.
8. The limited capability (LMTCPB) parameter on the Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command is set to *YES.

Monitoring Programs That Adopt the Owner's Authority

List all programs defined with adopted authority. For programs that run under the authority of users with special authority (for example, *ALLOBJ) do an analysis of the subprogram and library arrangement.

For a particular user, you can print all programs with adopted authority by using this command:

```
DSPPGMADP USRPRF(user-profile-name) OUTPUT(*PRINT)
```

Monitoring Job Descriptions

Review the job descriptions to see if a specific user profile is specified for jobs to run under. Use the Display Job Description (DSPJOBD) to look at the USER parameter to see if it specifies a specific user profile name. Analyze the authorities of the specific user profile.

Procedures for Monitoring Security Periodically

This topic describes some specific steps you can do at any site. All the commands shown include an OUTPUT(*PRINT) parameter. You will use the printout you obtain many times.

Step 1: Use the example commands in Figure 6-1. The first command produces a printout of all defined user profiles. The other commands produce a printout in considerably more detail for selected users.

```
DSPAUTUSR SEQ(*GRPPRF) OUTPUT(*PRINT)

DSPOBJD OBJ(*ALL) OBJTYPE(*USRPRF) OUTPUT(*PRINT)
        DETAIL(*BASIC)

DSPUSRPRF USRPRF(user-profile-name) TYPE(*BASIC) OUTPUT(*PRINT)

DSPUSRPRF USRPRF(user-profile-name) TYPE(*ALL) OUTPUT(*PRINT)
```

Figure 6-1. User Profile Inspection. Use these commands to inspect user profiles. Two levels of detail are produced by the two DSPUSRPRF commands.

Unless there is a particular reason for doing so, do not list the IBM-supplied user profiles in detail. Some of these profiles are very large because of the number of owned objects. The printouts are too large to be useful, and producing the printouts affects system performance. A substantial audit would probably list all of the actual user profiles in detail. A well planned site has few users with large profiles. Good planning involves group profiles, authorization lists, the use of the public authority parameter, and selective security control at the library level (rather than at the object level). User profiles with large numbers of authorities, appearing to be randomly spread over most of the system, can reflect a lack of security planning.

When examining user profiles, determine how many people have special authorities. The only general rule is that use of these authorities must be minimized. (The programmer at your site will most likely have more authority than others.) You should be certain that your site's owner or manager understands any exposures.

Step 2: Use the commands in Figure 6-2. The first command lists the names of all the libraries in the system. The other commands list the objects in the library and the authorities for the library.

```
DSPOBJD OBJ(*ALL) OBJTYPE(*LIB) OUTPUT(*PRINT)

DSPLIB LIB(library-name) OUTPUT(*PRINT)

DSPOBJAUT OBJ(library-name) OBJTYPE(*LIB)
          OUTPUT(*PRINT)
```

Figure 6-2. Library Inspection. You can use these CL commands to obtain an overview of system libraries.

The IBM-supplied system libraries are very large. It is common to add local objects to some of these libraries, and it may be necessary to obtain detailed displays of the system libraries. Obtain a printout showing the authorities for all libraries.

Step 3: This last step is to audit programs. It is not practical to inspect all the programs in the system. However, it is possible to examine certain aspects of all users with special authority. You can list all programs that are owned by these users and that adopt the owner's authority. In addition to random sampling of programs, there are categories of programs that you should inspect. Inspect any program that is owned by a user with *ALLOBJ special authority and that adopts the owner's authority. Verify the public authority of the program that adopts the owner's authority to see which users can call the program.

```
DSPPGMADP USRPRF(user-profile-name)  OUTPUT(*PRINT)

DSPOBJAUT OBJ(library-name/object-name)  OBJTYPE(*PGM) OUTPUT(*PRINT)

DSPOBJD OBJ(library-name/object-name)  OBJTYPE(*PGM) OUTPUT(*PRINT)
          DETAIL(*FULL)
```

Figure 6-3. Selected Program Inspection. Display the current authorities, the date the last change was made, and a description of a program (library-name) with these commands.

Monitoring Security Using History Log Commands

The following commands help isolate specific events from the history log. The general format of the command is:

```
DSPLOG LOG(QHST) PERIOD((start-time start-date)
       (end-time end-date))  MSGID(message-identifier)
       OUTPUT(*PRINT)
       {or OUTPUT(*)}
```

For example, the following command displays all the messages recorded in the history log for the current date. (The default start date is *CURRENT.)

```
DSPLOG LOG(QHST)
```

The following command displays all messages for December 1990 that have message numbers in the range CPF2201 to CPF22FF. Most security messages are in this range. The message number CPF2200 causes all messages in the range CPF2201 through CPF22FF to be selected.

```
DSPLOG LOG(QHST) PERIOD((*AVAIL 120190)(*AVAIL 123190))  MSGID(CPF2200)
```

The following command displays all the times that CPF2218 occurred starting from December 20, 1990 until the end of the log. The CPF2218 message indicates that someone tried to get an object file without the correct authority.

```
DSPLOG LOG(QHST) PERIOD((0000 122090)(*AVAIL *END)) MSGID(CPF2218)
```

When viewing messages from a display station (rather than printing them), place the cursor on a message, and press the Help key. Additional information such as the time and date of when the attempt was made is shown.

Monitoring the Security Officer's Actions

The following procedure can be used to monitor the use of commands used by the security officer or users with all object (*ALLOBJ) special authority.

1. Create a journal to log the security officer's actions. Journal entries cannot be deleted. Enter the following commands:

   ```
   CRTJRNRCV JRNRCV(SECLOG01)
     AUT(*EXCLUDE) TEXT('SECOFR receiver')
   CRTJRN JRN(SECLOG) JRNRCV(SECLOG01)
     AUT(*EXCLUDE) TEXT('Log SECOFR actions')
   ```

2. Make QCLSECOFR the initial program or menu option for the security officer.
3. Create the CL program QCLSECOFR as follows:

```
/*****************************************************************/
/* SECLOG - This program will present the Command Entry display     */
/*          and allow a user to issue commands.   The commands that */
/*          are run are recorded in the journal SECLOG.             */
/* OUTPUT - The entry in the journal receiver associated with the   */
/*          security log has three possible forms                   */
/*           *ERROR *  command - error detected when running        */
/*           *ENDRQS*  command - end request during command running */
/*           command  BLANK    - normal completion of command      */
/* RECOMMENDATIONS - Potential use is to record the actions of an   */
/*          *ALLOBJ user such as the QSECOFR.  To prevent the       */
/*          entry of commands on other command lines, define the    */
/*          user profile as LMTCPB(*YES).                           */
/*****************************************************************/
SECLOG:PGM
            DCL         VAR(&MSG) TYPE(*CHAR) LEN(512)
            DCL         VAR(&KEYVAR) TYPE(*CHAR) LEN(4)
            DCL         VAR(&RTNTYPE) TYPE(*CHAR) LEN(2)
RECEIVE:/*************************************************/
            /* Receiving a request message shows command entry.    */
            /*************************************************/
            RCVMSG      PGMQ(*EXT) MSGTYPE(*RQS) RMV(*NO) +
                          KEYVAR(&KEYVAR) MSG(&MSG) RTNTYPE(&RTNTYPE)
            MONMSG      MSGID(CPF2415) EXEC(RETURN)  /* F3(Return)  */
            /*   F4 - PROMPT RTNTYPE  = '10'     */
            IF    (&RTNTYPE = '10') +
                CHGVAR &MSG ('?' *CAT &MSG)  /* F4(Prompt)        */
            /*************************************************/
            /* Syntax check and allow prompting if requested.      */
            /*************************************************/
            CALL QCMDCHK (&MSG 512)
            MONMSG CPF0000  EXEC(GOTO RECEIVE)
            /*************************************************/
            /* Remove old request and replace with the new request.*/
            /* The new request includes any values from prompting. */
            /*************************************************/
            RMVMSG      PGMQ(*EXT) KEYVAR(&KEYVAR) CLEAR(*BYKEY)
            SNDPGMMSG TOPGMQ(*EXT) MSGTYPE(*RQS) MSG(&MSG)
            RCVMSG PGMQ(*EXT) MSGTYPE(*RQS) RMV(*NO)
            CALL QCMDEXC (&MSG 512) /* RUN CMD */
            /*************************************************/
            /* Issue command and log in SECLOG.                    */
            /*************************************************/
            MONMSG      MSGID(CPF1907) EXEC(CHGVAR &MSG +
                              ('*ENDRQS* ' || &MSG ) ) /* ENDRQS  */
            MONMSG      MSGID(CPF0000) EXEC(CHGVAR &MSG +
                              ('*ERROR* ' || &MSG ) )
            SNDJRNE JRN(SECLOG)  ENTDTA(&MSG)
            MONMSG CPF0000  EXEC(SIGNOFF *LIST)
            GOTO RECEIVE
           ENDPGM
```

4. Change the user profile to LMTCPB(*YES) so that commands can be entered only through the program that will log the requests.

System-Provided Security Auditing Using Journals

This topic describes the security auditing function provided by the system. The auditing function allows the security officer to monitor security by gathering data about specific security-related events.

The security auditing function is optional. You must take specific steps (described in "Setting Up Security Auditing" on page 6-11) to set up security auditing.

Security auditing shows the security events occurring on your system. The following events can be logged in the security auditing journal:

- Programs that use restricted instructions
- All programs that access objects using interfaces that are not supported
- Save and restore information
- Authorization failures
- Deleted objects
- Security related functions, including:
  - User profiles that submit jobs using a job description containing a user profile name for which the user does not have *USE authority
  - User profiles that sign on displays that have work station entries that reference job descriptions that have user profile names specified

Note: Logging of security-related information is done when a direct action by the user is taken. Security-related changes can occur internally on the system that will not be logged to the auditing journal (QAUDJRN). For example, when restoring authority for a user profile, many grant authority operations are done internally. Only one journal entry (type RU) is written to the QAUDJRN journal.

Security auditing statistics are recorded in the QAUDJRN journal in library QSYS using journal entries. The QAUDLVL value controls which security-related events are logged to the security auditing journal (QAUDJRN). One or more of the following values can be specified. If *NONE is specified, no other value can be specified.

***NONE:** No security-related events are logged to the QAUDJRN journal.

***AUTFAIL:** The system logs a journal entry (AF and PW) for each authorization failure that occurs.

***PGMFAIL:** The system logs a journal entry (AF) for each object domain, blocked instruction, or program validation value check failure.

***SAVRST:** The system logs a journal entry (RJ, RO, RA, RP, or RU) for each restore operation that includes:

- Job descriptions that contain user names
- Programs that adopt the authority of the owner's user profile
- Objects with ownership changes
- Objects with authority changes
- Authority for user profiles

***DELETE:** The system logs a journal entry (DO) for each delete operation.

Note: No journal entry is written to QAUDJRN journal for objects deleted in library QTEMP.

***SECURITY:** The system logs a journal entry (CA, CP, DS, OW, PA, SV, NA, PS, SE, or JD) for each security-related function that includes:

- Changing object authority (authorization list and objects)
- Creating, changing, and restoring user profiles
- Resetting the DST security officer password
- Changing object ownership
- Changing programs to adopt the owner's authority
- Changing system values
- Changing network attributes
- Changing subsystem routing
- Specifying a user profile name for the USER parameter on the Change Objects Description (CHGJOBD) command or the Create Job Description (CRTJOBD) command
- Generating a profile handle through the QSYSGETPH application program interface (API)
- Target user profile changed during pass-through

You should know how to perform journal management operations, such as saving a journal receiver, changing journal receivers, and deleting old journal receivers. For more information about changing, deleting, and saving journal receivers, see the topic "Saving and Deleting Auditing Journal Receivers" on page 6-44.

The *Database Guide* has more information about journals and journal receivers.

When you want to analyze the security data, the information in QAUDJRN journal can be analyzed using the Display Journal (DSPJRN) command. With this command, entries can be written to a database file. An application program or a utility, such as the query utility, can be used to analyze the data.

Setting Up Security Auditing

To set up security auditing, do the following:

1. Create a journal receiver in a library of your choice by using the Create Journal Receiver (CRTJRNRCV) command:

   ```
   CRTJRNRCV  JRNRCV(user-library-name/QAUDRCV1)
          AUT(*EXCLUDE) TEXT('Auditing Journal Receiver')
   ```

   Specifying *EXCLUDE on the AUT parameter limits access to the information stored in the journal.

   You should name the journal receiver AUDRCV1, or a similar name, such as QAUDRCV, or QAUDRCV3, which can be used to create a naming convention for future journal receivers.

   After you create the first receiver, you can create additional receivers and attach them to the QSYS/QAUDJRN journal automatically with the correct naming convention by using the Change Journal (CHGJRN JRNRCV(*GEN))

command. You will probably want to place the journal receiver in one of your libraries that is saved regularly.

2. Create the QSYS/QAUDJRN journal by using the Create Journal (CRTJRN) command. The name QSYS/QAUDJRN *must* be used, and you must have authority to add objects to QSYS. You need to specify the name of the journal receivers you created in the previous step and any other options on the command. You should also consider who you want to have authority to this journal.

   ```
   CRTJRN  JRN(QSYS/QAUDJRN) JRNRCV(user-library-name/QAUDRCV1)
         AUT(*EXCLUDE) TEXT('Auditing Journal')
   ```

3. Change the auditing level system value QAUDLVL using the Change System Value (CHGSYSVAL) command. Only a user with all object (*ALLOBJ) and security administrator (*SECADM) special authorities can change this system value.

   The VALUE parameter on the CHGSYSVAL command determines the level of auditing on the system. If *NONE is specified, no other value can be specified.

   The system requires that the QSYS/QAUDJRN journal be created before this system value is changed to request security auditing.

For information on processing the security journal entries, see the topic "Converting Security Auditing Journal Entries" on page 6-14.

QAUDJRN Journal

The auditing journal, QSYS/QAUDJRN, is processed like any other journal. Files can also be journaled to it, although it is recommended that you keep it solely for security information. You can use the Send Journal Entry (SNDJRNE) command to send other entries to this journal. While there are additional operational considerations involved in using several journals, there are advantages to *not* allowing any file entries in the QAUDJRN journal. System entries also appear in the journal QAUDJRN. These are the entries with a journal code of J, which relate to initial program load (IPL) and general operations performed on journal receivers (for example, a save of the receiver).

If damage occurs to the journal or to its current receiver so that the auditing entries cannot be journaled, a message is sent to the QSYSOPR message queue. The job trying to send the journal entry continues normally. Recovery from a damaged journal or journal receiver is the same as for other journals. See the *Backup and Recovery Guide* for recovery information.

If the journal and receiver reach a storage threshold, a message is sent to the QSYSOPR message queue indicating that the journal is reaching maximum size and some action must be taken.

You can use the OUTFILE parameter on the Display Journal (DSPJRN) command to write the auditing journal entries to a database file that you can process.

You can also use the Receive Journal Entry (RCVJRNE) command on the QAUDJRN journal to receive the entries as they are written to the QAUDJRN journal.

Journal Entry Types for QAUDJRN Journal

A **journal entry** is a record in a journal receiver that contains information about database files.

Each journal entry contains the standard prefix fields for any journal entry (for example, date, time, journal sequence number). See the topics "Using the Display Journal Command to Analyze the QAUDJRN Journal Data" on page 6-16 for more information about entry-specific data.

You can request the system to audit save and restore information, authorization failures, deleted objects, or security related functions. The system provides the following journal entries types:

Entry Type	Description
AF	All authority errors (object access): The authority failures journal entry provides data summarizing each failure to access objects because of insufficient authority or object domain violations.
CA	Changes to authorization list or object authority: The changes to authorization list or object authority journal entry provides data summarizing the objects and authorization lists that had changes made to their authority.
CP	Create, change, or restore of user profiles: The create, change, or restore of user profiles journal entry provides data summarizing each user profile that was created, changed, or restored.
DO	All delete operations: The delete operations journal entry provides data summarizing the objects that were deleted from the system. **Note:** No journal entry is written to QAUDJRN journal for objects deleted in library QTEMP.
DS	Dedicated service tools (DST) security officer password reset requested: The DST security officer password reset request journal entry provides data summarizing when a request was made to reset the DST security officer password to the system supplied default.
JD	Changes to the USER parameter on the Change Job Description (CHGJOBD) command journal entry provides data summarizing the job descriptions that had the USER parameter changed.
NA	Changes to network attributes: The change to network attributes journal entry provides data summarizing the network attributes that were changed.
OW	Changes to object ownership: The changes to object ownership journal entry provides data summarizing the objects that had their owner changed.
PA	Changes to Programs (CHGPGM) command that will adopt the owner's authority: The changes to program journal entry provides data summarizing the programs that were changed to specify that they use the owner's authority when they are called.
PS	Profile handle generation through the QSYSGETPH application program interface (API) and target user profile changed during pass-through.

PW Passwords that are not valid: The password journal entry provides data summarizing passwords that were used but were not valid or attempts to sign on using a user ID that does not exist.

RA Restore of objects when authority changes: The restore of objects when authority changes journal entry provides data summarizing each object that had changes to its authority when it was restored.

RJ Restore of job descriptions that contain user names: The restore of job descriptions that contain user names journal entry provides data summarizing each job description that contains a specific user profile name.

RO Restore of objects when ownership information changes: The restore of objects when ownership changes journal entry provides data summarizing each object that had its owner changed when it was restored.

RP Restore of programs that adopt the program owner's authority: The restore of programs that adopt the program owner's authority journal entry provides data summarizing each program that adopts the owner's authority that were restored on the system.

RU Restore of authority for user profiles: The restore of authority for user profiles journal entry provides data summarizing each user profile that had its authority restored.

SE Changes to subsystem routing: The change to subsystem routing journal entry provides data summarizing the subsystems that had their routing entry changed.

SV Changes to system values: The changes to system values Journal entry provides data summarizing the system values that were changed.

Converting Security Auditing Journal Entries

You can use the OUTFILE parameter on the Display Journal (DSPJRN) command to write the auditing journal entries into a database file that you can process. The OUTFILE parameter allows you to name a file or member. If the member exists, it is cleared before the records are written. If the member does not exist, it is added. If the file does not exist, a file is created using the record format QJORDJE. This format defines the standard heading fields for each journal entry, but the auditing data is defined as a single large field.

To avoid having to process the auditing data as a single large field, field reference files are supplied to help you process the auditing journal entries. Each journal entry type has a database file that maps the entire journal entry for this specific journal entry type.

The following table shows the physical file and message ID associated with each journal entry type.

Table 6-2. Physical Files Associated with Journal Entry Codes

Physical File Name	Journal Entry Code	Message ID	Description
QASYAFJE	AF	CPI2246 CPI2247 CPI2248 CPI2249 CPI2250 CPI2268 CPI2270 CPI2274	Authority violation Domain violation Submit job violation Default sign-on Object validation failure Blocked instruction violation Invalid profile handle Read-only storage violation
QASYCAJE	CA	CPI2253	Authority change
QASYCPJE	CP	CPI2266	User profile changed
QASYDOJE	DO	CPI2263	Delete of object
QASYDSJE	DS	CPI2267	Request to change DST QSECOFR password
QASYJDJE	JD	CPI2264	Job description change to specify user name
QASYNAJE	NA	CPI2257	Network attribute changes
QASYOWJE	OW	CPI2254	Ownership change
QASYPAJE	PA	CPI2255	Change program to adopt
QASYPSJE	PS	CPI2272 CPI2273	Profile handle generated Pass-through target user profile changed
QASYPWJE	PW	CPI2251 CPI2252	Password not valid User ID not valid
QASYRAJE	RA	CPI2261	Restore of object and authority changes
QASYRJJE	RJ	CPI2259	Restore of job description that contains a user profile name
QASYROJE	RO	CPI2260	Restore of object with ownership changes
QASYRPJE	RP	CPI2258	Restore of programs that adopt
QASYRUJE	RU	CPI2262	Restore of authority for user
QASYSEJE	SE	CPI2265	Routing entry changed
QASYSVJE	SV	CPI2256	System value change

You can control the use and selection criteria of the DSPJRN command so that you do not convert the same entries several times. For example, you can select all entries in a specific range of dates. You could convert all of the entries at a cutoff point for your auditing analysis (for example, monthly). One or more journal receivers may have been used during the month. Notice that each use of the DSPJRN command to the same member causes the member of the database file to be cleared before any new entries are added.

For an example of a program that creates output in a readable format, see the topic "Example Program for Analyzing the QAUDJRN Journal" on page 6-35.

Using the Display Journal Command to Analyze the QAUDJRN Journal Data

The Display Journal (DSPJRN) command allows you to analyze the data written to QAUDJRN journal. The outfile parameter (OUTFILE) allows you to write the journal entries or a subset of journal entries (based on journal entry code) to the outfile. The outfile information also includes:

- User name
- System name

Some of the security auditing information can also be accessed using the CPF22xx messages located in the QHST log. Security auditing data available through the QHST messages is a subset of the method described here. For more information on QHST messages, see the *CL Programmer's Guide*.

The Display Journal (DSPJRN) command displays the QAUDJRN journal. The journal entries in the QAUDJRN journal provides a record of security-related events. The following examples show the format of the journal displays. Following the display examples are tables with descriptions of the journal entries for each type of entry. All journal entry fields before the name of the job on the displays are used to create the header information on the Journal Entry display. There are seventeen types of QAUDJRN journal entries. These entries are in four categories:

Category	Definition
Authority failures	Authority failures include: • Profile handle used that is not valid • Authorization to object • Submit job • Default sign-on
Save and Restore	Restore information about programs that adopt, job descriptions containing a user profile name, object and ownership changes, object and authority changes, and restore authority for user profiles
Security	Security information that includes: • Profile handle generation • Changing object authority (authorization list and objects) • Changing pass-through target user profile • Creating, changing, and restoring user profiles • Resetting the DST security officer password • Changing object ownership • Changing programs to adopt the owner's authority • Changing system values • Changing network attributes • Changing subsystem routing

- Specifying a user profile name for the USER parameter on the Change Job Description (CHGJOBD) or the Create Job Description (CRTJOBD) command

Program failures Program failure information that includes:

- Using instructions blocked by the system in a program
- Program-restore validation errors
- Read-only storage violation
- Object-domain violations

Delete Deleting of objects

Other entries may also appear in the QAUDJRN journal. You may ignore them for purposes of QAUDJRN journal analysis.

The following command will show the Display Journal Entries display:

```
===> DSPJRN QAUDJRN
```

The following display is an example of the Display Journal Entries display:

```
                          Display Journal Entries

Journal . . . . . :   QAUDJRN        Library . . . . . :   QSYS

Type options, press Enter.
  5=Display entire entry

Opt    Sequence  Code  Type  Object      Library     Job         Time
          28018   J     PR                          UEHLINGS1   11:02:05
          28020   T     AF                          QSYSARB     11:07:33
          28021   T     PW                          QINTER      11:08:18
          28022   T     AF                          QSYSARB     11:09:29
          28023   T     AF                          QSYSARB     11:10:07
          28024   T     AF                          QSYSARB     11:10:32
          28025   T     AF                          QSYSARB     11:32:57
 5        28026   T     PW                          QINTER      11:58:05
          28027   T     PW                          BEUCH       11:58:43
          28028   T     PW                          QINTER      12:37:34
          28029   T     PW                          QINTER      12:37:36
          28030   T     PW                          QINTER      12:49:04    +

F3=Exit   F12=Cancel
```

The following journal entry is shown by typing a 5 in the *Opt* column before the number 280286 in the *Sequence* column and pressing the Enter key. Information from all fields through the *Library* column is used on the following display:

```
                          Display Journal Entry

Journal . . . . . :   QAUDJRN        Library . . . . . :   QSYS
Sequence . . . . . :  28026

Code . . . . . :      T  - Audit trail entry
Type . . . . . :      PW - Invalid password or user ID

Object . . . . . :                   Library . . . . . :
Member . . . . . :

Position to  . . . . .               (Column)

           Entry specific data
Column     *...+....1....+....2....+....3....+....4....+....5
00001     'PBECHER    DSP03                                  '
00051     ' '

                                                                  Bottom
Press Enter to continue.

F3=Exit   F6=Display only entry specific data
F10=Display only entry details   F12=Cancel   F24=More keys
```

If you want to display only entry details, press F10 (Display only entry details). The following display is shown.

```
                     Display Journal Entry Details

Journal . . . . . :   QAUDJRN        Library . . . . . :   QSYS
Sequence . . . . . :  28026

Code . . . . . :      T  - Audit trail entry
Type . . . . . :      PW - Invalid password or user ID

Object . . . . . :                   Library . . . . . :
Member . . . . . :                   Flag . . . . . :      0
Date . . . . . :      02/08/90       Time . . . . . :      11:58:05
Count/RRN  . . . . :  0              Program . . . . . :   QWTMCMNL

Job  . . . . . :      023225/QSYS/QINTER
User profile . . . :  QSYS
Commit cycle ID  . . :  0



Press Enter to continue.

F3=Exit   F10=Display entry   F12=Cancel   F14=Display previous entry
F15=Display only entry specific data
```

To display the entry, press F10 (Display entry). The following display is shown.

```
                          Display Journal Entry

Journal . . . . . :   QAUDJRN        Library . . . . . :   QSYS
Sequence . . . . . :  28026

Code . . . . . :      T  - Audit trail entry
Type . . . . . :      PW - Invalid password or user ID

Object . . . . . :                   Library . . . . . :
Member . . . . . :

Position to  . . . . .               (Column)

           Entry specific data
Column     *...+....1....+....2....+....3....+....4....+....5
00001     'PBECHER    DSP03                                  '
00051     ' '
                                                                  Bottom
Press Enter to continue.

F3=Exit   F6=Display only entry specific data
F10=Display only entry details   F12=Cancel   F24=More keys
```

You can also display the entry-specific data in hexadecimal format. To display the hexadecimal format, press F24 (More keys) and then press F11 (Hexadecimal format.) The entry-specific data changes to hexadecimal format as shown on the following display. To return to the character data, press F11 again.

```
                          Display Journal Entry

Journal . . . . . :   QAUDJRN        Library . . . . . :   QSYS
Sequence . . . . . :  28026

Code . . . . . :      T  - Audit trail entry
Type . . . . . :      PW - Invalid password or user ID

Object . . . . . :                   Library . . . . . :
Member . . . . . :

Position to  . . . . .               (Column)

           Entry specific data
Column     *  . . . + . . . . 1 . . . . + . . . . 2 . . . . +
00001     'D7C2C5C3C8C5D940404040C4E2D7F0F3404040404040404040'
00026     '4040404040404040404040404040404040404040404040404040'
00051     '40'
                                                                  Bottom
Press Enter to continue.

F3=Exit   F6=Display only entry specific data
F10=Display only entry details   F12=Cancel   F24=More keys
```

This display does not have space for all the data in hexadecimal format; therefore, you must press the Page Down key to display the remaining entry-specific data. A + in the lower right corner of the display indicates additional data.

Entry-Specific Data for QAUDJRN Journal

The following tables describe the entry-specific data for the QAUDJRN journal entries.

Format for Authority Failure Journal Entries (AF)

The following is the database description of the records for the journal entry type AF, which represents the entry for authority failures. This record is contained in physical file QASYAFJE, which is a part of the QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always AF for authority failures.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Violation Type	Char(1)	The type of violation. **A**= An attempt was made to access an object or perform an operation that the user was not authorized to. **B**= A program ran a restricted machine interface instruction. **C**= A program, which failed the restore-time program validation checks, was restored. See validation value violation type at offset 157 for more information. **D**= A program accessed an object through an unsupported interface or callable program not documented as a callable API. **J**= An attempt was made to submit a job under a job description which has a user profile specified, and the submitter of the job was not authorized to that user profile. **S**= An attempt was made to sign on without entering a user ID or password. **R**= An attempt was made to update an object that was defined as read-only. **P**= An attempt was made to use a profile handle that is not valid on the QWTSETP API. See "Security Level 40 Considerations" on page 2-5 for more information.

Offset	Field	Format	Description
157	Validation Value Violation Type	Char(1)	The type of cyclic redundancy check (Validation Value). **A**= Changed object was restored that may violate security **B**= Object restore and all authority revoked **C**= Validation Value failure on program. Copy of program that was translated was restored **D**= A changed object was restored as requested by the security officer **E**= System install time error detected
158	Object Name	Char(10)	The name of the object.
168	Library Name	Char(10)	The name of the library the object is in.
178	Object Type	Char(8)	The type of object.
186	Job Name	Char(10)	The name of the job.
196	User Name	Char(10)	The job user name.
206	Job Number	Zoned(6,0)	The job number.
212	Program Name	Char(10)	The name of the program.
222	Program Library	Char(10)	The name of the library where the program is found.
232	User Profile	Char(10)	The name of the user using the program.
242	Work Station Name	Char(10)	The name of the work station or work station type.
253	Program Instruction Number	Zoned(7,0)	The instruction number of the program.
257	(Reserved Area)	Char(16)	
272	Office User	Char(10)	The name of the office user.
282	Folder or Document Name	Char(12)	The name of the document or folder.
284	(Reserved Area)	Char(8)	
302	Folder Path	Char(63)	The path of the folder.
365	Office on Behalf of User	Char(10)	User working on behalf of another user.

Format for Authority Changes Journal Entries (CA)

The following is the database description of the record for the journal entry type CA, which represents the entry for authority changes. This record is contained in physical file QASYCAJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always CA for authority change-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.

Offset	Field	Format	Description
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Changes to authority
157	Object Name	Char(10)	The name of the object.
167	Library Name	Char(10)	The name of the library the object is in.
177	Object Type	Char(8)	The type of object.
185	User Name	Char(10)	The job user name.
195	Authorization List Name	Char(10)	The name of the authorization list.
205	Object Existence	Char(1)	**Y**=*OBJEXIST
206	Object Operational	Char(1)	**Y**=*OBJOPR
207	Object Management	Char(1)	**Y**=*OBJMGT
208	Authorization List Management	Char(1)	**Y**=*AUTLMGT
209	Authorization List	Char(1)	**Y**=*AUTL authority
210	Read Authority	Char(1)	**Y**=READ authority
211	Add Authority	Char(1)	**Y**=*ADD
212	Update Authority	Char(1)	**Y**=*UPD
213	Delete Authority	Char(1)	**Y**=*DLT
214	Exclude Authority	Char(1)	**Y**=*EXCLUDE
215	(Reserved Area)	Char(7)	
222	Command Type	Char(3)	The type of command used. **GRT**= Grant **RVK**= Revoke
225	(Reserved Area)	Char(20)	
245	Office User	Char(10)	The name of the office user.
255	Folder or Document Name	Char(12)	The name of the document or folder.
267	(Reserved Area)	Char(8)	
275	Folder Path	Char(63)	The path of the folder.
338	Office on Behalf of User	Char(10)	User working on behalf of another user.
348	Personal Status	Char(1)	**Y**=Personal status changed
349	Access Code	Char(1)	**A**= Access code changed **R**= Access code removed
350	Access Code	Char(4)	Access code.

Format for Changes to User Profiles Journal Entries (CP)

The following is the database description of the record for the journal entry type CP, which represents the entry for changes to user profiles. This record is contained in physical file QASYCPJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always CP for change profile-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Change to a user profile
157	User Profile Name	Char(10)	The name of the user profile that was changed.
167	Library Name	Char(10)	The name of the library.
177	Object Type	Char(8)	The type of object.
185	Command Name	Char(3)	The type of command used. **CRT**= CRTUSRPRF **CHG**= CHGUSRPRF **RST**= RSTUSRPRF **DST**= CHGDSTPWD
188	Password Changed	Char(1)	**Y**= Password changed
189	Password *NONE	Char(1)	The name of the authorization list.
190	Password Expired	Char(1)	**Y**=Password expired
191	*ALLOBJ Special Authority	Char(1)	**Y**=*ALLOBJ special authority
192	Job Control Special Authority	Char(1)	**Y**=*JOBCTL special authority
193	Save System Special Authority	Char(1)	**Y**=*SAVSYS special authority
194	Security Administrator Special Authority	Char(1)	**Y**=*SECADM special authority
195	Spool Control Special Authority	Char(1)	**Y**=*SPLCTL special authority
196	Service Special Authority	Char(1)	**Y**=*SERVICE special authority
197	(Reserved Area)	Char(15)	
212	Group Profile	Char(10)	The name of a group profile.
222	Owner	Char(10)	Owner of objects created as a member of a group profile.
232	Group Authority	Char(10)	Group profile authority.
242	Initial Program	Char(10)	The name of the user's initial program.
252	Initial Program Library	Char(10)	The name of the library where the initial program is found.
262	Initial Menu	Char(10)	The name of the user's initial menu.

Offset	Field	Format	Description
272	Initial Menu Library	Char(10)	The name of the library where the initial menu is found.
282	Current Library	Char(10)	The name of the user's current library.
292	Limited Capabilities	Char(10)	The value of limited capabilities parameter.
302	User Class	Char(10)	The user class of the user.
312	Priority Limit	Char(1)	The value of the priority limit parameter.
313	Profile Status	Char(10)	User profile status.

Format for Delete of an Object Journal Entries (DO)

The following is the database description of the record for the journal entry type DO, which represents the entry for objects that are deleted. This record is contained in physical file QASYDOJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Note: No journal entry is written to QAUDJRN for objects deleted in library QTEMP.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always DO for delete of object-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Object was deleted
157	Object Name	Char(10)	The name of the object.
167	Library Name	Char(10)	The name of the library the object is in.
177	Object Type	Char(8)	The type of object.
185	(Reserved Area)	Char(20)	
205	Office User	Char(10)	The name of the office user.
215	Folder or Document Name	Char(12)	The name of the document or folder.
227	(Reserved Area)	Char(8)	
235	Folder Path	Char(63)	The path of the folder.
298	Office on Behalf of User	Char(10)	User working on behalf of another user.

Format for DST Password Reset Journal Entries (DS)

The following is the database description of the record for the journal entry type DS, which represents the entry for resetting the DST password. This record is contained in physical file QASYDSJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always DS for DST-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry.
			A= Reset of DST password
157	DST Password Reset	Char(1)	**Y**=Request to reset DST password.

Format for Change of USER Parameter of a Job Description Journal Entries (JD)

The following is the database description of the record for the journal entry type JD, which represents the entry for changes to the USER parameter of a job description. This record is contained in physical file QASYJDJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always JD for job description QAUDJRN-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.

Offset	Field	Format	Description
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= User profile specified for the USER parameter of a job description
157	Command Type	Char(3)	The type of command used. **CHG**= Change Job Description (CHGJOBD) command. **CRT**= Create Job Description (CRTJOBD) command.
160	Job Description	Char(10)	The name of the job description that had the USER parameter changed.
170	Library Name	Char(10)	The name of the library the object is in.
180	Object Type	Char(8)	The type of object.
188	Old User	Char(10)	The name of the user profile specified for the USER parameter before the job description was changed.
198	New User	Char(10)	The name of the user profile specified for the user parameter when the job description was changed.

Format for Network Attribute Changes Journal Entries (NA)

The following is the database description of the record for the journal entry type NA, which represents the entry for changes to network attributes. This record is contained in physical file QASYNAJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always NA for network attribute-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Change to network attribute.
157	Network Attribute	Char(10)	The name of the network attribute.
167	New Network Attribute Value	Char(250)	The value of the network attribute after it was changed.

Offset	Field	Format	Description
417	Old Network Attribute Value	Char(250)	The value of the network attribute before it was changed.

Format for Ownership Changes Journal Entries (OW)

The following is the database description of the record for the journal entry type OW, which represents the entry for changes to ownership. This record is contained in physical file QASYOWJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always OW for ownership-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Change of object owner
157	Object Name	Char(10)	The name of the object.
167	Library Name	Char(10)	The name of the library the object is in.
177	Object Type	Char(8)	The type of object.
185	Old Owner	Char(10)	Old owner of the object.
195	New Owner	Char(10)	New owner of the object.
205	(Reserved Area)	Char(20)	
225	Office User	Char(10)	The name of the office user.
235	Folder or Document Name	Char(12)	The name of the document or folder.
247	(Reserved Area)	Char(8)	
255	Folder Path	Char(63)	The path of the folder.
318	Office on Behalf of User	Char(10)	User working on behalf of another user.

Format for Change Program to Adopt Owners Authority Journal Entries (PA)

The following is the database description of the record for the journal entry type PA, which represents the entry for programs that were changed to adopt their owner's authority. This record is contained in physical file QASYPAJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always PA for program adopt-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Change program to adopt owner's authority
157	Program Name	Char(10)	The name of the program.
167	Program Library	Char(10)	The name of the library where the program is found.
177	Object Type	Char(8)	The type of object.
185	Owner	Char(10)	The name of the owner.

Format for Profile Swap Journal Entries (PS)

The following is the database description of the record for the journal entry type PS, which represents the entry for user profile handle generated by the QSYGETPH API and represents the entry for the target user profile changed during pass-through. This record is contained in physical file QASYPSJE, which is part of the QAUDJRN journal and can be used by a programmer to create a program that formats entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always RP for restoring a program-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.

Offset	Field	Format	Description
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry.
			A= Profile swap during pass-through.
			H= Profile handle generated by the QSYGETPH API.
157	User Profile	Char(10)	User profile name.
167	Source Location	Char(8)	Pass-through source location.
175	Original target user profile	Char(10)	Original pass-through target user profile.
185	New target user profile	Char(10)	New pass-through target user profile.

Format for Password and User ID Journal Entries (PW)

The following is the database description of the record for the journal entry type PW, which represents the entry for password and user ID errors. This record is contained in physical file QASYPWJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always PW for password-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Violation Entry Type	Char(1)	The type of violation
			P= Password not valid
			U= User name not valid
157	User Name	Char(10)	The job user name.

Offset	Field	Format	Description
167	Device name	Char(10)	The name of the device where the password or user ID was entered.

Format for Restore of Object and Authority Changes Journal Entries (RA)

The following is the database description of the record for the journal entry type RA, which represents the entry for restoring objects that have their authority changed. This record is contained in physical file QASYRAJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always RA for restore of object authority-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Changes to authority for object restored
157	Object Name	Char(10)	The name of the object.
167	Library Name	Char(10)	The name of the library the object is in.
177	Object Type	Char(8)	The type of object.
185	Authorization List Name	Char(10)	The name of the authorization list.
195	Public Authority	Char(1)	**Y**=Public authority set to *EXCLUDE.
196	Private Authority	Char(1)	**Y**=Private authority removed.
197	AUTL Removed	Char(1)	**Y**=Authorization list removed from object.
198	(Reserved Area)	Char(20)	
218	Folder or Document Name	Char(12)	The name of the document or folder.
230	(Reserved Area)	Char(8)	
238	Folder Path	Char(63)	The path of the folder.

Format for Restore of Job Descriptions Journal Entries (RJ)

The following is the database description of the record for the journal entry type RJ, which represents the entry for restoring job descriptions that specify a user profile name in the USER parameter. This record is contained in physical file QASYRJJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always RJ for restore of job description-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Restore of a job description that had a user profile specified in the USER parameter.
157	Job Description Name	Char(10)	The name of the job description restored.
167	Library Name	Char(10)	The name of the library the job description was restored to.
177	Object Type	Char(8)	The type of object.
185	User Name	Char(10)	The name of the user profile specified in the job description.

Format for Restore of Object and Ownership Changes Journal Entries (RO)

The following is the database description of the record for the journal entry type RO, which represents the entry for restoring objects that have their ownership changed. This record is contained in physical file QASYROJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always RO for restore of object ownership-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.

Offset	Field	Format	Description
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry. **A**= Restore of objects that had ownership changed when restored
157	Object Name	Char(10)	The name of the object.
167	Library Name	Char(10)	The name of the library the object is in.
177	Object Type	Char(8)	The type of object.
185	Old Owner	Char(10)	The name of the owner before ownership was changed.
195	New Owner	Char(10)	The name of the owner after ownership was changed.
205	(Reserved Area)	Char(20)	
225	Folder or Document Name	Char(12)	The name of the document or folder.
237	(Reserved Area)	Char(8)	
245	Folder Path	Char(63)	The path of the folder.

Format for Restore of Programs that Adopt Journal Entries (RP)

The following is the database description of the record for the journal entry type RP, which represents the entry for restoring programs that adopt the owner's authority. This record is contained in physical file QASYRPJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always RP for restore of program-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	

Offset	Field	Format	Description
156	Entry Type	Char(1)	The type of entry.
			A= Restore of programs that adopt the owner's authority
157	Program Name	Char(10)	The name of the program.
167	Program Library	Char(10)	The name of the library where the program is found.
177	Object Type	Char(8)	The type of object.
185	Owner Name	Char(10)	Name of the owner.

Format for Restore Authority for User Profiles Journal Entries (RU)

The following is the database description of the record for the journal entry type RU, which represents the entry for restoring authority for user profiles. This record is contained in physical file QASYRUJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always RU for restore of authority for users event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry.
			A= Restore of authority to user profiles
157	User Name	Char(10)	The name of the user profile that was restored.
167	Library Name	Char(10)	The name of the library.
177	Object Type	Char(8)	The type of object.

Format for Change of Subsystem Routing Entry Journal Entries (SE)

The following is the database description of the record for the journal entry type SE, which represents the entry for changes to routing entries for subsystems. This record is contained in physical file QASYSEJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always SE for routing entry QAUDJRN-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry.
			A= Subsystem routing entry changed.
157	Subsystem Name	Char(10)	The name of the object.
167	Library Name	Char(10)	The name of the library the object is in.
177	Object Type	Char(8)	The type of object.
185	Program Name	Char(10)	The name of the program that changed the routing entry.
195	Library Name	Char(10)	The name of the library for the program.
205	Sequence Number	Char(4)	The sequence number.
209	Command Name	Char(3)	The type of command used.
			ADD= ADDRTGE
			CHG= CHGRTGE
			RMV= RMVRTGE

Format for System Value Changes Journal Entries (SV)

The following is the database description of the record for the journal entry type xx, which represents the entry for changes to system values. This record is contained in physical file QASYSVJE, which is a part of QAUDJRN journal and can be used by a programmer to create a program that formats the entries.

Offset	Field	Format	Description
1	Length of Entry	Zoned(5,0)	Total length of the journal entry, including the entry length field.
6	Sequence Number	Zoned(10,0)	Applied to each journal entry. Initially set to 1 for each new or restored journal. Reset to 1 when a new receiver is attached.
16	Journal Code	Char(1)	Always T.
17	Entry Type	Char(2)	Always SV for system value-logged event.
19	Date of Entry	Char(6)	The system date that the entry was made.
25	Time of Entry	Zoned(6,0)	The system time that the entry was made.
31	Name of Job	Char(10)	The name of the job that caused the entry to be generated.
41	User Name	Char(10)	The user profile name associated with the job.

Offset	Field	Format	Description
51	Job Number	Zoned(6,0)	The job number.
57	Program Name	Char(10)	The name of the program that made the journal entry.
67	(Reserved Area)	Char(51)	
118	User Profile	Char(10)	The name of the current user profile.
128	System Name	Char(8)	The name of the system.
136	(Reserved Area)	Char(20)	
156	Entry Type	Char(1)	The type of entry.
			A= Change to system values
157	System Value	Char (10)	The name of the system value.
167	New System Value	Char(250)	The value the system value was changed to.
417	Old System Value	Char(250)	The value of the system value before it was changed.

Example Program for Analyzing the QAUDJRN Journal

The following example program creates a readable version of the QAUDJRN journal. This example program TPSAUDLOG1 can be found in library QUSRTOOL and requires the PRINT tool that is also found in that library.

Messages CPI2246 through CPI2274, shipped in the message file QCPFMSG, give a text description of the entries in the QAUDJRN journal.

Display Audit Log (DSPAUDLOG) Command

The Display Audit Log (DSPAUDLOG) command displays the entries from the security audit journal (QAUDJRN) by retrieving messages which make the entries in the auditing journal readable by the end user.

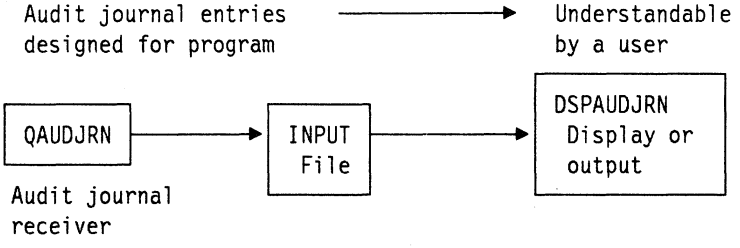
This is useful for:

- Reviewing messages in the security journal
- Printing a history of security violations

A typical command would be entered as:

```
DSPAUDLOG   OPTION(*CURRENT)  OUTPUT(*)
```

This command retrieves messages that can translate the information in the auditing journal into a form that can be read by a user.



Display Audit Log (DSPAUDLOG) Command Parameters

The following is a description of the parameters used for the Display Audit Log (DSPAUDLOG) command.

OPTION

Selection of the journal entries

***CURRENT:** Printer or display the entries from the journal receiver currently attached.

**file-name:* Printer or display entries from a file that has been previously created by the Display Journal (DSPJRN) command. This option should be used if other than the current journal is desired or selection by entry type Specify ENTDTALEN(357) to be sure that a 512 byte record is created.

```
/*******************************************/
/* The entry length must be 512            */
/* HEADER + ENTRY Length = Message Data    */
/*  155   +  357   =   512                 */
/*******************************************/
 DSPJRN    JRN(QAUDJRN) JRNCDE(T) ENTDTALEN(357) +
           OUTPUT(*OUTFILE) OUTFILFMT(*TYPE2) +
           OUTFILE(QTEMP/AUDITJRN)
```

OUTPUT

The member to be checked.

*: If interactive, display the information at workstation. If batch, print the first level text.

***PRINT:** Print first level text.

***SECLCL** Print first and second level text.

Installing the Display Audit Log (DSPAUDLOG) Command

Do the following to install the program:

1. Create a library to be used for the objects. If it is an existing library, then this step can be omitted. The characters *user-library* are used to represent the library in the following commands.

   ```
   CRTLIB   user-library
   ```

2. Create the install program in the library:

   ```
   CRTCLPGM   PGM(user-library/TPSAINST) SRCFILE(QUSRTOOL/QATTCL)
   ```

3. Run the install program, preferably by submitting a batch job. One parameter must be supplied, which is the name of the *user-library* where objects will be created. This library must exist at run time.

   Note: The PRINT tool found in QUSRTOOL is needed by the DSPAUDLOG tool.

   ```
   SBMJOB     CMD(CALL PGM(sec-library/TPSAINST) PARM(user-library))
   ```

The following objects will be created in the specified library.

User Library			QUSRTOOL Library	
Object	**Type**	**Attribute**	**Source Member**	**Source File**
DSPAUDLOG	*CMD		DSPAUDLOG	QATTCMD
TPSAUDLOG1	*PGM	CLP	TPSAUDLOG1	QATTCL

The following program creates the command, file, and program needed by the DSPAUDLOG command.

```
000100000000/*************************************************************/
000200900106/* Name:  TPSAINST - Create of DISPLAY AUDIT LOG command   */
000300900106/*        creates the command , file , and program needed  */
000400900106/*        by the DSPAUDLOG (DISPLAY AUDIT LOG) command.    */
000500000000/*                                                         */
000501900106/*            OBJECT      TYPE    SOURCE      MEMBER      */
000502900106/*            ------      ----    --------    --------    */
000503900106/*            INPUT      *FILE      --          --        */
000504900106/*            DSPAUDLOG  *CMD     QATTCMD     DSPAUDLOG   */
000505900106/*            TPSAUDLOG1 *PGM     QATTCL      TPSAUDLOG1  */
000506900106/*                                                         */
000507900106/*                                                         */
000600000000/* Parameters                                              */
000700000000/*         IN  Library Name                                */
000800000000/*              The library must exist before calling this */
000900000000/*              program.                                   */
001000000000/*                                                         */
001100900106/* Inovcation:     CALL TPSAINST  lib-name                 */
001200000000/*************************************************************/
001300000000/*************************************************************/
001400000000 TGRINST:    PGM        PARM(&USERLIB)
001500000000
001600000000             DCL        VAR(&USERLIB)  TYPE(*CHAR)  LEN(10)
001700000000             DCL        &ERRORSW *LGL /* Std err*/
001800000000             DCL        &MSGID *CHAR LEN(7) /*Std err*/
001900000000             DCL        &MSGDTA *CHAR LEN(100) /* Std err*/
002000000000             DCL        &MSGF *CHAR LEN(10) /* Std err*/
002100000000             DCL        &MSGFLIB *CHAR LEN(10) /* Std err*/
002200000000             MONMSG     MSGID(CPF0000) EXEC(GOTO STDERR1)/*Std err*/
```

```
002300000000  /* ****************START OF PROGRAM   ***************/
002400000000             CHKOBJ     OBJ(QSYS/&USERLIB) OBJTYPE(*LIB)
002500000000             MONMSG     MSGID(CPF9800) EXEC(DO) /* No library */
002600000000             SNDPGMMSG  MSGID(CPF9898) MSGF(QCPFMSG)
                                    MSGTYPE(*ESCAPE) +
002700000000                          MSGDTA('The library must exist +
002800000000                          to create the objects')
002900000000             ENDDO      /* No library */
003000000000/*  ******** CREATE COMMANDS *********/
003100891226 DSPAUDLOG:   DLTCMD     CMD(&USERLIB/DSPAUDLOG)
003200000000             MONMSG     MSGID(CPF0000)
003300891226             CRTCMD     CMD(&USERLIB/DSPAUDLOG) +
003400900106                          PGM(&USERLIB/TPSAUDLOG1) +
003500900106                          SRCFILE(QUSRTOOL/QATTCMD) +
003600900106                          SRCMBR(DSPAUDLOG ) TEXT(*SRCMBRTXT) +
003700891226                          VLDCKR(*NONE) MODE(*ALL) ALLOW(*ALL) +
003800891226                          MAXPOS(1)
006700000000/*  ******** CREATE FILES    ********                      */
006800891226             DLTF       FILE(QTEMP/INPUT)
006900000000             MONMSG     MSGID(CPF0000)
007000900106             CRTPF      FILE(QTEMP/INPUT) RCDLEN(512) +
007100891226                          TEXT(*SRCMBRTXT) MAXMBRS(1) AUT(*USE)
007101900206/*  ******** OVERIDE FOR CRTPGM *****                      */
007102900106             OVRDBF     FILE(INPUT) TOFILE(QTEMP/INPUT)
008900000000/*  ******** CREATE PROGRAMS ********                      */
009000900106 TPSADDLOG1: DLTPGM     PGM(&USERLIB/TPSAUDLOG1  )
009100000000             MONMSG     MSGID(CPF0000)
009200900106             CRTCLPGM   PGM(&USERLIB/TPSAUDLOG1  ) +
009300891226                          SRCFILE(QUSRTOOL/QATTCL) TEXT(*SRCMBRTXT)
009400891226
009401891226/*  ******** Delete Temporary File***                      */
009402891226             DLTF       FILE(QTEMP/INPUT)
009403891226             MONMSG     MSGID(CPF0000)
013500000000             RETURN     /* Normal end of program */
013600000000/*********************************************************************
*/
013700000000 STDERR1:               /* Standard error handling routine */
013800000000             IF         &ERRORSW SNDPGMMSG MSGID(CPF9999) +
013900000000                          MSGF(QCPFMSG) MSGTYPE(*ESCAPE) /* Func chk
*/
014000000000             CHGVAR     &ERRORSW '1' /* Set to fail ir error occurs
*/
014100000000 STDERR2:    RCVMSG     MSGTYPE(*DIAG) MSGDTA(&MSGDTA) MSGID(&MSGID)
+
014200000000                          MSGF(&MSGF) MSGFLIB(&MSGFLIB)
014300000000             IF         (&MSGID *EQ '       ') GOTO STDERR3
014400000000             SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGFLIB/&MSGF) +
014500000000                          MSGDTA(&MSGDTA) MSGTYPE(*DIAG)
014600000000             GOTO       STDERR2 /*Loop back for addl diagnostics*/
014700000000 STDERR3:    RCVMSG     MSGTYPE(*EXCP) MSGDTA(&MSGDTA) MSGID(&MSGID)
+
014800000000                          MSGF(&MSGF) MSGFLIB(&MSGFLIB)
014900000000              SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGFLIB/&MSGF) +
015000000000                          MSGDTA(&MSGDTA) MSGTYPE(*ESCAPE)
015100000000              ENDPGM
```

The following creates the Display Audit Log (DSPAUDLOG) command:

```
000200891209/*********************************************************/
000300891226 /* Command Name: DSPAUDLOG - Display Audit Log command */
000301891226 /*                                                       */
000302891226 /* Command Syntax:                                       */
000303891226 /*   DSPAUDLOG   OPTION(*CURRENT or file-name)           */
000304891226 /*               OUTPUT(* or *PRINT or *SECLVL)          */
000400891226 /*                                                       */
000500891226 /* Command Processing Program: TPSAUDLG                  */
001100891209 /*********************************************************/
001200891226        CMD        PROMPT('Display Audit Log')
001300891226        PARM       KWD(OPTION) TYPE(Q1) DFT(*CURRENT) +
001400891226                     SNGVAL((*CURRENT)) PROMPT('Entry selection +
001500891226                     file . . . .')
001600891209        PARM       KWD(OUTPUT) TYPE(*CHAR) LEN(1) RSTD(*YES) +
001700891221                      DFT(*) SPCVAL((*PRINT P) (*SECLVL S) (* *)) +
001800891209                       PROMPT('OUTPUT . . . . . . . . . . .')
001900891222 Q1:    QUAL       TYPE(*NAME) LEN(10)
002000891222        QUAL       TYPE(*NAME) LEN(10) DFT(*LIBL) SPCVAL((*LIBL) +
002100891222                     ('*CURLIB'))
```

The following program prints or displays the QAUDJRN data using messages to interpret the journal entries.

```
PGM (&PARM1  &PARM2)
/*****************************************************************/
/* TPSAUDLOG1-  This program prints or displays the audit journal data */
/*            using messages to interpret the journal entries so the */
/*            log can be understood by a user.                       */
/*            The input parameters allow the user to select the      */
/*            form of the input and output.                          */
/*                                                                   */
/*     PARM1  Determines the source of the journal entries.          */
/*            *CURRENT - Print/Display the entries from the journal  */
/*                       receiver currently attached.                */
/*            file-name -Print/Display entries from a file that has  */
/*                       been created by the DSPJRN command.  This   */
/*                       option may be used if the user wants to be  */
/*                       selective in the entries based on time or   */
/*                       type.                                       */
/*     PARM2  Determines form used to present the journal entries.   */
/*            *          Display the journal entries on the screen   */
/*                       in the joblog if interactive.  In batch,    */
/*                       this is equivalent to the *PRINT option.    */
/*            *PRINT     -Print first level test for journal entries */
/*            *SECLVL    -Print both first and second level text.    */
/*                       Option may be used if the user wants to be  */
/*                       selective in the entries based on time or   */
/*                       type.                                       */
/*                                                                   */
/* Program Logic                                                     */
/*   1. If *CURRENT journal receiver is requested, then use the      */
/*      command DSPJRN to load entries into a data base file.        */
/*     If a user named file is provided, override to read from the   */
/*      file.                                                        */
/*   2. Loop reading entries from the file                           */
/*     a. Look up the associated message based upon entry type.      */
/*     b. Use journal entry as message data to produce a human       */
/*        readable form of the information.                          */
/*     c. Print or display the information based upon the second     */
/*        parameter.                                                 */
/*****************************************************************/
        DCL     &PARM1     *CHAR  20      /*Journal receiver option */
                                          /* *CURRENT--do a DSPJRN  */
                                          /* filename--user file    */
        DCL     &PARM2     *CHAR  1       /*LISTING OPTIONS         */
                                          /* *------- DISPLAY       */
                                          /* P (*PRINT) PRINT 1 LVL */
                                          /* S (*SECLVL) PRINT 1&2  */
        DCL     &DISPLAY   *LGL  VALUE('1') /* OUTPUT indicator     */
                                          /* 1=display 0=print      */
        DCL     &JOBTYPE   *CHAR  1       /* 0=Batch  1=Interactive */
        DCL     &DATSEP    *CHAR  1       /* DATE SEPARATOR         */
        DCL     &SYSNAME   *CHAR  8       /*System name for report  */
        DCL     &JRN       *CHAR  10      /*Journal name or *CURRENT*/
        DCL     &JRNLIB    *CHAR  10      /*Journal library name    */
        DCL     &MSGID     *CHAR  7       /*Message ID              */
        DCL     &MSGDTA    *CHAR  50      /*ERROR Handling          */
        DCL     &MSGF      *CHAR  10      /*ERROR Handling          */
        DCL     &MSGLIB    *CHAR  10      /*ERROR Handling          */
        DCL     &RTNTYPE   *CHAR  2       /*ERROR Handling          */
        DCL     &ERROR     *LGL           /*ERROR Handling          */
        DCL     &ENTTYP    *CHAR  4       /*Entry Type              */
        DCL     &I         *DEC  (3 0)    /*Index used to search    */
        DCL     &MSG       *CHAR  120     /*Retrieved first level   */
                                          /*text                    */
        DCL     &SECLVL    *CHAR  2000    /*Retrieved second level  */
                                          /*text                    */
        DCL     &SECLVLLEN *DEC  (5 0)    /*Length of second level  */
        DCL     &LINE      *CHAR  132     /*Formatted print line    */
        DCL     &STARTPOS  *DEC  (5 0)    /*SCAN start point        */
        DCL     &SCANLEN   *DEC  (5 0)    /*SCAN length             */
        DCL     &STRLEN    *DEC  (5 0)    /*SCAN string length      */
        DCL     &FOUND     *DEC  (5 0)    /*SCAN new line offset    */
        DCL     &MOVE      *DEC  (5 0)    /*Length to move          */
```

```
/*************************************************************/
/* The following table is used to associate a message ID     */
/* with a journal entry type.                                */
/*************************************************************/
DCL    &TABLE      *CHAR   319   VALUE('+
         TAFACPI2246+
         TAFBCPI2268+
         TAFCCPI2250+
         TAFDCPI2247+
         TAFJCPI2248+
         TAFSCPI2249+
         TAFPCPI2270+
         TAFRCPI2274+
         TCAACPI2253+
         TCPACPI2266+
         TDOACPI2263+
         TDSACPI2267+
         TJDACPI2264+
         TNAACPI2257+
         TOWACPI2254+
         TPAACPI2255+
         TPSACIP2273+
         TPSHCPI2272+
         TRAACPI2261+
         TSEACPI2265+
         TRJACPI2259+
         TROACPI2260+
         TRPACPI2258+
         TRUACPI2262+
         TOWACPI2254+
         TPWPCPI2251+
         TPWUCPI2252+
         TSVACPI2256+
         XXXXNOTHERE')
    DCLF       FILE(INPUT)
MONMSG     CPF0000 EXEC(GOTO ERROR)
/********  START OF PROGRAM  ***************/
/*************************************************************/
/* Initialization and determine the source of journal        */
/* entries.                                                  */
/*************************************************************/
    RTVJOBA    TYPE(&JOBTYPE) DATSEP(&DATSEP)
IF (&PARM2 *NE '*' *OR &JOBTYPE *EQ '0') DO
     CHGVAR     &DISPLAY '0'
     OVRPRTF    FILE(QPRINT) USRDTA(AUDITRPT) +
                  SPLFNAME(DSPAUDLOG)
    RTVNETA    SYSNAME(&SYSNAME)
    PRINT      ACTION(*OPN) TITLE('Audit Log      System +
                 Name: ' || &SYSNAME)
  ENDDO
CHGVAR &JRN     %SST(&PARM1 1 10)
CHGVAR &JRNLIB  %SST(&PARM1 11 10)
IF (&JRN = '*CURRENT') DO
    /*****************************************/
    /* The maximum length entry is limited by */
    /* the 512 limit on message data.         */
    /* This will result in truncation of the  */
    /* entries for changes to system values   */
    /* and network attributes.                */
    /* HEADER + ENTRY Length = Message Data   */
    /*  155   +  357   =   512                */
    /*****************************************/
    DSPJRN     JRN(QAUDJRN) JRNCDE(T) ENTDTALEN(357) +
                 OUTPUT(*OUTFILE) OUTFILFMT(*TYPE2) +
                 OUTFILE(QTEMP/AUDITJRN)
  OVRDBF     INPUT  QTEMP/AUDITJRN
ENDDO
ELSE   OVRDBF     FILE(INPUT) TOFILE(&JRNLIB/&JRN)
/*************************************************************/
```

```
            /*************************************************************/
            /* Main loop of program to read and process each entry       */
            /*************************************************************/
            /*************************************************************/
READ:       RCVF
            MONMSG      MSGID(CPF0864) EXEC(GOTO CMDLBL(EOF))
            CHGVAR &ENTTYP ( %SST(&INPUT 16 3) || %SST(&INPUT 156 1) )
            /*************************************************************/
            /* Search entry table to match entry type to corresponding   */
            /* message ID.   Some entries have more detail messages      */
            /* based upon fields in the journal entry                    */
            /*************************************************************/
            CHGVAR    &I    1
LOOP:       IF (%SST(&TABLE &I 4) = 'XXXX') GOTO MATCH /*end of table   */
            IF (%SST(&TABLE &I 4) = &ENTTYP) GOTO MATCH
            CHGVAR  &I (&I+11)
            GOTO LOOP
MATCH:      CHGVAR &I     (&I + 4)
            CHGVAR &MSGID  %SST(&TABLE &I   7)
            /*************************************************************/
            /*The OUTPUT parameter is used to determine if a printed     */
            /*or display or printed output is desired.                   */
            /*  If printed:                                              */
            /*    1 Retrieve message text using journal entry as message*/
            /*      data.                                                */
            /*    2 Print first level text.                              */
            /*    3 If second level text is desired format the print     */
            /*      lines by scanning for new line characters.           */
            /*  If display:                                              */
            /*    Send the message to the job using the journal entry    */
            /*    as message data.   The message handler handles the     */
            /*    formatting of second level text.                       */
            /*************************************************************/
            IF (&DISPLAY)                 /*DISPLAYED output*/+
                SNDPGMMSG  MSGID(&MSGID) MSGF(QCPFMSG) MSGDTA(&INPUT) +
                             TOPGMQ(*PRV)
            /*************************************************************/
            /* The bulk of this code is used to produce the printed      */
            /* report and format second level text.                      */
            /*************************************************************/
            ELSE DO
                RTVMSG      MSGID(&MSGID) MSGF(QCPFMSG) MSGDTA(&INPUT) +
                              MSG(&MSG) SECLVL(&SECLVL) +
                              SECLVLLEN(&SECLVLLEN)
                CHGVAR      &LINE (%SST(&INPUT 19 2) || &DATSEP +
                                 ||  %SST(&INPUT 21 2) || &DATSEP +
                                 ||  %SST(&INPUT 23 2)        +
                               *BCAT %SST(&INPUT 25 2) || ':' +
                                 ||  %SST(&INPUT 27 2) *BCAT &MSG  )
                PRINT       LINE(&LINE)
```

```
            /*********************************************************/
            /*The following code prints the second level text.       */
            /*The text is formatted by scanning for (&N OR &P)       */
            /*control characters in the retrieved message.           */
            /*********************************************************/
            IF (&PARM2 *EQ 'S') DO
               CHGVAR &STARTPOS   1
SECLVL:        CHGVAR &SCANLEN (&SECLVLLEN - &STARTPOS)
               IF (&SCANLEN *LE 0) GOTO ENDSECLVL
               IF ( &SCANLEN *GT 120 )  CHGVAR &SCANLEN 120
               CHGVAR &STRLEN (&SCANLEN + &STARTPOS)
               CHGVAR &FOUND &STARTPOS
SCAN:          IF (&FOUND *GT &STRLEN  ) DO
                  CHGVAR &FOUND 0
                  GOTO SCANEXIT
               ENDDO
               IF ( (%SST(&SECLVL &FOUND 2) *EQ '&N') *OR +
                    (%SST(&SECLVL &FOUND 2) *EQ '&P')) /* NULL */
               ELSE DO
                 CHGVAR &FOUND (&FOUND +1)
                 GOTO SCAN
               ENDDO
SCANEXIT:
               IF (&STARTPOS = &FOUND) DO
                 CHGVAR     &STARTPOS (&FOUND    + 2)
                 GOTO       SECLVL
               ENDDO
               IF (&FOUND = 0) DO     /* REACHED END OF SCAN NO & */
                 CHGVAR     &LINE ('             '  +
                                   ||  %SST(&SECLVL &STARTPOS  120)  )
                 CHGVAR     &STARTPOS (&STARTPOS + 120)
               ENDDO
               ELSE DO               /* FOUND MOVE PARTIAL STRING */
                 CHGVAR     &MOVE (&FOUND - &STARTPOS)
                 CHGVAR     &LINE ('             ' || +
                        %SST(&SECLVL &STARTPOS &MOVE))
                 CHGVAR     &STARTPOS (&FOUND + 2)
               ENDDO
              PRINT      LINE(&LINE)
              GOTO       SECLVL
            ENDDO
ENDSECLVL:
          ENDDO
         GOTO READ

         /*********************************************************/
         /* END OF MAIN PROGRAM LOGIC                             */
         /*********************************************************/
  EOF:   RCVMSG     MSGTYPE(*EXCP)   /* NORMAL EXIT PROCESSING   */
         IF ( *NOT &DISPLAY )   PRINT  ACTION(*CLO)
         GOTO EXIT
 ERROR:  /*********************************************************/
         /* ERROR EXIT FOR PROGRAM                                */
         /*********************************************************/
              IF &ERROR  GOTO EXIT
              CHGVAR     &ERROR '1'
              IF ( *NOT &DISPLAY )   PRINT  ACTION(*CLO)
 RECEIVE:     RCVMSG     MSGTYPE(*ANY) MSGDTA(&MSGDTA) MSGID(&MSGID) +
                           RTNTYPE(&RTNTYPE) MSGF(&MSGF) +
                           MSGFLIB(&MSGLIB)
              IF    (&RTNTYPE *NE '15') /* NOT ESCAPE MESSAGE    */ +
                DO
                 SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGF) MSGDTA(&MSGDTA) +
                           MSGTYPE(*DIAG)
                  GOTO RECEIVE
                ENDDO
              SNDPGMMSG  MSGID(&MSGID) MSGF(&MSGF) MSGDTA(&MSGDTA) +
                           MSGTYPE(*ESCAPE)
 EXIT:        ENDPGM
```

Saving and Deleting Auditing Journal Receivers

It is recommended that large journal receivers should be detached periodically, saved, and then deleted to free storage.

Before a journal receiver can be deleted, use the Change Journal (CHGJRN) command to detach it from the journal.

When a new journal receiver is attached to the journal, the previously attached journal receiver is automatically detached. You can change the journal and delete the journal receiver periodically when the journal receiver becomes too large without affecting normal system operations and without affecting the user of the files. However, it is recommended that this operation not be performed during the time the system is at maximum use.

When you specify JRNRCV(*GEN) on the CHGJRN command, the system creates the new receiver with the same values as the currently attached receiver, and in the same library. These values include the owner, public authority, unit or ASP identifier, threshold, and text.

When you change from one receiver to another (when JRN(*GEN) is specified), the last number in the new receiver name is one greater than the last sequence number in the detached receiver. For example, if you detach journal receiver AUDRCV0001, the system created journal receiver is AUDRCV0002.

In the following example, the journal receiver AUDRCV0001 is being detached from journal QAUDJRN.

The Work Journal Attribute (WRKJRNA) command can be used to determine which receivers are currently active and which files are being journaled, and to provide a list of all receivers associated with the journal. You can display all journals on the system by doing the following:

1. Type the following on any command line and press the Enter key:

   ```
   WRKJRN
   ```

2. On the Select Journal Name display, press F4 (Prompt) to display all the journals on the system.
3. If you want to display the status of a specific journal, type 5 (Display journal status) in the *Opt* column and press the Enter key.
4. All system-supplied journals start with the letter *Q*. To display the name of the journal receiver, type 5 in the *Option* field and press the Enter key.
5. Type the following command to detach the AUDRVC0001 journal receiver from the journal and create a new journal receiver named AUDRCV0002.

   ```
   CHGJRN JRN(QAUDJRN) JRNRCV(*GEN)
   ```

6. If you want to save the detached journal receiver, type the following:

   ```
   SAVOBJ OBJ(AUDRCV0001) OBJTYPE(*JRNRCV) LIB(USERLIB)
          DEV(TAP01)
   ```

7. It is recommended that you save the journal receiver. However, if you *do not* want to save the journal receiver and you do want to delete it, type the following:

   ```
   DLTJRNRCV JRNRCV(USERLIB/AUDRCV0001)
   ```

If you decided not to save the journal receiver and want to delete it, enter an I to ignore when the following message is shown:

```
Receiver not fully saved. ( C I )
```

8. Press the Enter key.

For more information about managing journals and journal receivers, see the *Backup and Recovery Guide*.

Chapter 7. Security Recommendations and Planning

The purpose of this chapter is to provide some guidelines and recommendations for planning. This chapter includes planning examples and example forms to help you become familiar with security commands and parameters.

Note: Security planning and implementation is the responsibility of the customer.

AS/400 Security Recommendations

The following topics provide security recommendations. These should be useful when planning a new AS/400 system.

The primary word here is **planning**. Effective security depends on establishing rules and guidelines for the new system.

Programmers

It is recommended that a plan for programmers be defined before any additional security planning. AS/400 systems are unusual in that many sites do not have (or need) programmers because the users purchase the available licensed programs rather than write their own. If there are no programmers, then this consideration does not apply.

Due to the nature of their tasks, programmers can usually go around any system security in one way or another. From their point of view, this is appropriate if it helps in their jobs. The programmer's tasks are typically:

- Installing the system
- Tuning the system (for performance)
- Installing the product
- Debugging and problem assistance for other users
- Assisting with installing complex application programs

Using security and enforcing it is often not seen as an important part of a programmer's job. To do their own tasks better or more conveniently they sometimes bypass correct security procedures. Bypassing the system security procedures might allow other users to bypass system security.

One solution is to make security the highest priority of the programmer. Unfortunately, this is difficult to do because this tends to conflict with day-to-day responsibilities of the programmer.

If the situation permits it, having the programmer also be the security officer can be a solution. In this case the programmer is not tempted to conveniently bypass the system security because he already has *ALLOBJ authority when signing on as the security officer. However, this breaks a basic security principle of separation of duties. Bypassing security will probably be due to:

- Lack of planning
- Intentional actions by an authorized user

For application programmers, the following should be considered:

- Do not grant *ALLOBJ, *SERVICE, or *SECADM special authority to the programmer.
- Use test libraries and prevent access to production libraries.
- Create programmer libraries and use a program that adopts authority to copy selected production data to programmer libraries for testing.

Naming Conventions

Naming conventions are important. The system does not enforce or control any naming conventions for objects. It is totally the user's responsibility.

It is recommended that you establish naming conventions. Effective security is not possible without this regardless of the size of the site. Until a convention is well established and understood, someone must enforce it manually. This is done by printing a list of the object names added to the system and inspecting the names.

The main objective of the naming convention is to have object names (especially group profiles and files) that are self-descriptive.

Naming Conventions for Users and Groups

A common convention is to use a combination of first and last names to create user profile names. The user's full name (and address) should be placed in the user description (TEXT) part of the profile when the user is defined. Avoid user names such as USER1, TEST, or anything similar.

Do not start any user name with the character Q. This letter is used by IBM for many system functions and other parts of the system.

Group profiles should always have a name that indicates it is a group profile. For example, DEPT977 would be a descriptive group name. A name like PLANNING is descriptive but does not distinguish whether it is a group or an individual user profile.

Some thought should be given to profile (and object) names. Many AS/400 commands support generic names. For example, DEPT* (when used for an object or profile name) would find all names beginning with the letters DEPT. There is no operation to find, for example, all names ending with DEPT. If group profile names for all departments begin with DEP or DEPT, it is easy to refer to all group profiles. Likewise, if all object names for the payroll department begin with PAY, it is easy to refer to these objects in a single command.

Naming Conventions for Objects

A self-identifying name should, if possible, indicate the purpose of the object (a library, a file, a command), the owner, the user, or the project that uses the object. The purpose of planning is to develop a set of rules for creating names because new names are constantly needed and created.

It is recommended that you establish a naming convention as a major part of any AS/400 system. While the naming convention should apply to all objects, it is most important for library, program, and file names. The plan for any new project should include a plan for naming the objects involved. This plan should be described in the project documentation.

Text Descriptions of Objects

An object has a field for a text description of the object. The owner of the object is responsible for placing a clear and accurate description of the object in this field. It is important to make use of this field and to place meaningful information there. This field does not create a security risk because this information is only available to users that are given access to the object.

Protection Strategies

Different strategies for doing security are available on the AS/400 system:

- Library security
- Object security
- Menu security

These different approaches are described below.

Library Security

Library security establishes security at the library level. Library security assumes that libraries contain objects with similar protection requirements and that, in general, a nonspecific protection is adequate. Library security typically applies when application programs are maintained in separate libraries and that both test and production objects are separated at the library level.

Object Security

Object security defines authorization at the specific object level, for example, a file in a library. It is used where different objects within a library have different protection requirements. Object security may be necessary when the arrangement of the library does not reflect security requirements or can be used as an exception to the general authorization rules.

Menu Security

Menu security is related to limiting a user's capabilities and restricting him to a predefined secured environment. The user's initial program and menu will restrict him to the functions and objects he is allowed to use.

Recommendations

It is recommended that you use different methods in combination. You should use the following steps when developing the overall security strategy:

- Library security

  Libraries should be designed in a way that objects contained in a library have identical or at least similar protection requirements. Authorizations to libraries should then be established as a first step. It is recommended that specific authority be defined for all production libraries; it may be acceptable to cover test libraries through public authority only.

- Object security

  Specific object authority should only be defined to handle exceptions; otherwise, the default public authority should be used. Exceptions exist where only a few objects within a library have more critical protection requirements than defined for the library, and where temporary access must be given.

- Menu security

  It is recommended that you use the limited capability approach where appropriate with library and object security. This recommendation is based on the fact that library and object security are enforced by the system, while initial programs, menus, and so on, are largely user-designed and, therefore, more likely to be at risk.

Protection Techniques

The following topics describe different protection techniques you can use on the AS/400 system.

Authorization List and Group Profile Considerations

To simplify and reduce the number of authorizations on the system, you can use either an authorization list or a group profile. When a large number of authorizations exist on the system, authorizations become more difficult to manage, and the time it takes to save or restore the system increases.

Authorization Lists

When you add a user to an authorization list, you specify the authority for that user, such as *CHANGE or *USE authority. This authority is used when the user accesses all objects secured by the authorization list. Each user on the list is assigned authority independently. Therefore, a user on the list may be authorized differently from other users on the list. Any authority given specifically to a user for an object overrides the authority of the authorization list securing the object.

The advantage of an authorization list is that when an object that is secured by an authorization list is saved, the association of the object to the list is saved. If the object is deleted and then restored, the object is automatically linked again to the authorization list.

A user can be on many authorization lists, but an object can be secured by only one authorization list.

Group Profiles

A member of a group profile can share all the authorities given specifically to the group. For example, if the group profile is specified on an authorization list, the member gets the authority specified on the authorization list for the group. Any authority given specifically to the member overrides the authority of the authorization list and the group profile.

When an object is saved, the group profile is not affected. If the object is deleted and then restored, authority for the object must be granted to the group again if they are not the owner of the object.

See "Authority Checking" on page 4-29 for more information about how the system verifies a member's authority to an object.

A user can be a member of only one group profile but the group can have authority to multiple objects. For an example, see Figure 4-2 on page 4-10.

Security Consideration

If you plan to use group profiles, select a naming convention for the group profile such as GROUPXXX or DEPTXXX that allows you to identify the profile as a group profile. Then, when you are displaying the list of authorized users for an object, it is easier to identify that there are group profiles with additional users authorized to the object. This naming convention should be used for all group profiles.

The following is a comparison of authorization lists and group profiles.

Table 7-1. Authorization List and Group Profile Comparison

Item Being Compared	Authorization List	Group Profiles
Secure multiple objects	Yes	Yes
User can belong to more than one	Yes	No
Private authority overrides other authority	Yes	Yes
User is assigned authority independently	Yes	No
The authorities specified are the same for all objects	Yes	No
The object can be secured by more than one	No	Yes
Authority can be specified when the object is created	Yes	No
Can secure all object types	No	Yes
Association with object is deleted when the object is deleted	Yes	Yes
Association with object is saved when the object is saved	Yes	No

Individual versus Group Authorization

It is recommended that you use groups profiles for job functions and use group authority as a general rule. Individual users should only be given specific authority as an exception to the rule or be given temporary access to objects.

Authorization Lists

It is recommended that you use authorization lists where possible. They offer performance advantages over specific object authority and have the advantage that the authority is not lost when the objects secured by the authorization list are deleted.

Group Profiles

In an environment that is not planned, a group profile often starts as a user profile belonging to the lead programmer, or possibly the first programmer or user in a particular department. This initial user eventually acquires all the authorities and objects (files, and so on) needed for his particular project. If additional users or programmers are then added to the project, it may be convenient to use the first person's profile as the group profile. It is possible to do this, and the first person can continue using his profile (which is now also a group profile). However, this situation is not acceptable from an accountability point of view. In this situation, it is almost impossible to distinguish the actions and effects of the individual user from those of members using the group profile.

Therefore, it is recommended that group profiles be established as soon as possible after the system is installed. Any project or department should be considered for a group profile. Do not use individual profiles as group profiles.

Object Ownership

Planning any new AS/400 system, large or small, should include a plan for ownership of objects. This applies to all objects that are used by more that one person or are for production purposes. If ownership of objects is not controlled, confusion results.

In general, it is recommended that you use ownership by a group profile for files and other objects, whenever it is reasonable, before you use individual ownership. However, all members of the group have ownership control of all objects owned by the group.

Another advantage of group ownership is that new users that are added to a group immediately have access to all the objects for which the group has authority. Likewise, when a user is removed from a group he no longer has access to the objects for which the group profile has authority.

Logical Files

For access to critical files, logical files should be used. Then, the owner of the file can authorize other users to specific fields (for example, address and telephone number, but not salary) or specific records (for example, amounts less than $500) instead of the entire physical file.

Public Authority

Using public authority and authorization lists can greatly reduce the need for specific authority (individual or group). If a file, for example, is not confidential, the public authority should probably be *USE authority, depending on the type of the data. *USE authority allows anyone to read it, without needing any more security controls. If everyone in the organization needs to read the file, this is much better than making the public authority *EXCLUDE, and giving specific authority to anyone who needs to read it. The objective is to avoid unnecessary large lists of authorities in user profiles or authorization lists.

There is a tendency to overly restrict public authority (or its equivalent in other systems). It is recommended that the public authority for objects in any planned project or system be considered as carefully as all other security aspects. The default public authority for most objects is *CHANGE.

A common security strategy is to provide a reasonable public authority for almost all objects in a library. (Only more sensitive objects would have public authority specified as *EXCLUDE.) The library itself (because the library is also on object) provides the basic level of control. Thus very few of the objects in the library require specific authority. With a single authorization (to the library), a user gains access to almost everything in the library. This is the basis of *library security*. It is an effective way to reduce the total number of authorities in the system, which improves system performance especially for save and restore operations.

Adopted Authority

Adopted authority, although useful, can lead to security exposures. Programmers must be especially careful with this function. A routine, using the adopted authority of any user with *ALLOBJ special authority, can call any program and have it run with *ALLOBJ special authority. For example, a short, simple program, when owned by the security officer profile, is an open door into the system.

It is recommended that you do not allow any program to be installed that is owned by a user profile that has *ALLOBJ special authority unless the function of the program is very clear.

No one should be allowed to routinely operate with *ALLOBJ special authority or to create programs that adopt a profile with *ALLOBJ special authority unless his responsibility is clearly stated.

IBM-Supplied User Profiles

A number of user profiles are supplied by IBM when the system is shipped. These profiles are used as the owners of the system resources when you first install the system, when licensed programs are added, and during maintenance.

There is no password for most of these user profiles, and they cannot be used to sign on the system. The IBM-supplied user profiles and ownership of system resources should not be changed. However, you should change the passwords for the following user profiles:

- QSECOFR security officer
- QSRV full service functions (display/alter)
- QSRVBAS basic service functions
- QPGMR programmer
- QUSER work station user
- QSYSOPR system operator

The security officer's password is a somewhat special case. The normal password management rules apply, of course. In addition, the system provides an option for recovering or resetting the security officer's password.

Planning for Security

The purpose of this topic is to show you how to plan for security on your system. When you have completed planning users' authorities for objects, user profiles, and authorization lists, you can use the information to set up the group profiles, individual user profiles, and authorization lists on the system.

The planning you do now can save you time when you create the user profiles and when you grant authority to users for system resources.

Some of the decisions that you should make are to:

- Determine if you want system security
- Select who has responsibility for security
- Determine the types of security you want
- Establish a schedule for activating security
- Train operators on security

- Decide if you want any other security support

To plan effectively, you must understand the security concepts discussed previously in this manual.

Determine If You Want System Security

This topic provides information to help you determine if you need:

- Physical security
- Data security
- System-level security

Physical Security

You or someone in your business should have already decided what physical security to use when you planned for site security. Many of these security measures are repeated here for your review.

- You can control who can access the physical equipment by placing the system in a locked room.
- You can control the use of the entire system by using the keylock switch on the control panel. You can also limit the use of one or more display stations with an optional lock on the display station. These locks prevent the use of specific devices unless the key is in the unlocked position.
- You can protect your data and programs by copying them to tape or diskette. These copies allow you to quickly recover from the loss of your data due to accidental or intentional damage.

  The contents of these tapes and diskettes are not secured. You should allow only authorized people to access your tapes or diskettes.
- You can protect your tapes and diskettes by storing them in a fireproof safe or at another location. You should always keep your tapes and diskettes away from magnetic fields, such as those created by large electric motors.

Data Security

Data security is provided as part of the system. Data security helps you prevent unauthorized people from:

- Signing on your system.
- Using confidential information in the system (for example, payroll).
- Deleting or changing information. If records are deleted, important information can be lost before the error is discovered and corrected.

System-Level Security

System-level security is provided to prevent programs from accessing objects using interfaces that are not supported. The following information identifies what the system will do at different levels of security:

All security levels. The system:

- Writes a journal entry to the auditing journal each time a restricted instruction is used.

  This includes attempts to directly call system programs not documented at the call level interfaces or attempts to access internal system structures through pointer capabilities of languages such as C, PASCAL, assemblers,

and so forth. For example, directly calling the command processing program for the SIGNOFF command would cause a journal entry. See Appendix E, "Supported Call Level Interfaces" on page E-1 for a list of supported interfaces.

- Writes a journal entry to the auditing journal for each attempt to access objects using interfaces that are not supported
- Restricts programs that contain restricted instructions from compiling.

Security level 30 or above. The system writes a journal entry to the auditing journal when:

- User profiles submit jobs using a job description that contains a user profile name for which the user does not have *USE authority.
- User profiles sign on by pressing the Enter key at the Sign On display. The work station entry for the user references a job description that has a user profile name specified for the USER parameter.

Security Level 40. At security level 40:

- Attempts to access objects through interfaces that are not supported fail and a journal entry is written to the QAUDJRN journal.

  This includes attempts to directly call system programs not documented at the call level interfaces or attempts to access internal system structures through pointer capabilities of languages such as C, PASCAL, assemblers, and so forth. For example, directly calling the command processing program for the SIGNOFF command would cause a journal entry. For a list of call-level interfaces that are supported, see Appendix E, "Supported Call Level Interfaces" on page E-1.
- Program that contains restricted instructions will not compile. User cannot create programs containing any restricted instructions.
- Users submitting jobs using a job description containing a user profile name must have *USE authority to that user profile
- Sign-on by pressing the Enter key at the Sign On display fails. A valid user ID and password must be entered.
- The system will check for a validation value when restoring programs to the system. If the program does not have a validation value, or the validation fails, the program is translated again. See "Limiting the Restore of Programs That are Not Valid or Were Changed" on page 7-10 for more information about the validation value.

  If the translation fails (no template exists, a restricted instruction is used), the program is restored, but all public and private authorities are revoked and the ownership is transferred to QDFTOWN. This action is logged in the security journal, as a message in the job log, and in the restore report (if one is requested).
- User profiles on the system have special authorities set, based on the user class of the user profile. These special authorities are set to the same values as user profiles at security level 30. Changing from a level 30 to a level 40 system results in no change to special authorities on any user profile. The same is true when changing from security level 40 to security level 30.

Security Consideration

Security level 40 is recommended **after** your system has run at security level 30 or below. You should monitor the security journal at level 30 or below for violations, and change any programs that have errors before operating your critical applications at security level 40.

Limiting the Restore of Programs That are Not Valid or Were Changed

When the system security level is 40, the restore of programs that appear to have been modified or that contain restricted instructions is controlled by the allow object differences (ALWOBJDIF) parameter on the restore commands.

To detect changes to a program (rather than through a normal system interface) a validation value is calculated by the translator when the program is created. When the program is restored, the validation value is calculated again and compared to the validation value that was calculated when the program was created.

What happens next depends on the following:

- If the program was created on a previous release system, it will not have a validation value saved with it. The QSECURITY system value and the ALWOBJDIF parameter on the restore command will be checked as follows:
  - For security levels 10 through 30, or when ALWOBJDIF(*ALL) is specified, the program is restored with no further validation or notification given.
  - For security level 40, when ALWOBJDIF(*NONE) (the default) is specified, an attempt is made to translate the program again. If the translation succeeds, the copy of the program that has translated again is restored. No *AUTFAIL type journal entry is written in the QAUDJRN journal.

    If the translation fails, the original copy of the program is restored and all public and private authorities are revoked. Ownership of the program is transferred to QDFTOWN user profile. A *AUTFAIL type journal entry is written in the QAUDJRN journal indicating the changes. Message CPF375B is sent to the job log.
- If the program has a correct validation value saved with it, the original program is restored. The QSECURITY system value and the ALWOBJDIF parameter on the restore commands does not apply. No *AUTFAIL type journal entry is written in the QAUDJRN journal.
- If the program has a validation value that fails, an attempt is made to translate the program again.
  - If the translation succeeds, the program is restored. A *AUTFAIL type journal entry is written in the QAUDJRN journal indicating what happened. Message CPF375C is sent to the job log.
  - If the translation fails, the original copy of the program is restored from the media. The security and logging actions taken depend on the QSECURITY system value and the ALWOBJDIF parameter on the restore command.
    - For security levels 30 and below, a *AUTFAIL type journal entry is written to the QAUDJRN journal indicating that a program was restored that may result in a possible violation of security. Message CPF375A is sent to the job log.

- If the security level is 40, with ALWOBJDIF(*ALL) specified on the restore command, a *AUTFAIL type journal entry is written in the QAUDJRN journal indicating that a program was restored (as requested by a user with *ALLOBJ special authority) that may result in a possible violation of security. Message CPF375D is sent to the job log.
- If the security level is 40, with ALWOBJDIF(*NONE) specified on the restore command, all public and private authorities are revoked and ownership of the program is transferred to the QDFTOWN user profile. A *AUTFAIL type journal entry is written in the QAUDJRN journal indicating the changes. Message CPF375B is sent to the job log.

Table 7-2 shows what happens at each security level.

Table 7-2. Restore of Programs That Use Interfaces That Are Not Supported

Validation Value	Successful Translation	Level 40 or ALWOBJDIF (*NONE)	Level 40 ALWOBJDIF (*ALL)	Levels 30 and Below
None	Yes	1	N/A	N/A
None	No	2		
None	Not attempted	N/A	3	3
Valid	Not attempted	4	4	4
Not valid	Yes	5	5	5
Not valid	No	2	6	6

Notes:

1 Restore a copy of translated program. No notification given.

2 Restore the original program, revoke all public and private authorities, unlink any authorization lists, and change owner to QDFTOWN. Write journal entry to QAUDJRN journal and send a message to the job log.

3 Restore the original program. No notification given.

4 Restore the original program. No notification given.

5 Restore the translated program. Write journal entry to QAUDJRN journal and send a message to the job log.

6 Restore original program. Write journal entry to QAUDJRN journal and send a message to the job log.

Select Who Has Responsibility for Resource Security

The person who has overall responsibility for security should be aware of the information in your business that needs to be protected and the degree to which the information needs to be protected. This person is responsible for:

- Deciding what types of resource security are needed.
- Setting up and maintaining security.
- Training users as to the importance of security.
- Deciding if any other security support provided by the system is needed.

Determine the Types of Resource Security to Use

This topic explains the types of resource security that are available as part of the system. Resource security can help you protect your data and programs from unauthorized users.

Planning authority to resources on the system, such as files, libraries, and devices, allows you to determine which resources a user will have access to. When resources security is not active on the system, any user can sign on and can use any resource on the system.

To plan authority to resources, you must:

- Determine which resources contain sensitive information.
- Identify which users should be allowed to use those resources.
- Determine if you want to secure multiple resources with an authorization list or group profiles.

Determine the System Values to Use

A number of system values help you tailor security on your system. The following are descriptions of the system values you can use. These values can be changed using the Change System Values (CHGSYSVAL) command or the Work with System Value (WRKSYSVAL) command. For more information about using this command, see the topic "Working with System Values That Affect Security" on page 8-10.

In the following system value descriptions, some of the values that can be specified are underlined. The values that are underlined are the system-supplied defaults.

Changing the Security Auditing Level

The QAUDLVL value controls which security-related events are logged to the security auditing journal (QAUDJRN). One or more of the following values can be specified. If *NONE is specified, no other value can be specified.

***NONE:** No security-related events are logged to the QAUDJRN journal.

***AUTFAIL:** The system logs a journal entry (AF and PW) for each authorization failure that occurs.

***PGMFAIL:**

- Object-domain violations
- Blocked-instruction violations
- Validation-value errors
- Read-only storage violations

***SAVRST:** The system logs a journal entry (RJ, RO, RA, RP, or RU) for each restore operation that includes:

- Job descriptions that contain user names
- Programs that adopt the authority of the owner's user profile
- Objects with ownership changes
- Objects with authority changes
- Restore of authority for user profiles

***DELETE:** The system logs a journal entry (DO) for each delete operation.

Note: No journal entry is written to QAUDJRN journal for objects deleted in library QTEMP.

***SECURITY:** The system logs a journal entry (CA, CP, DR, OW, PA, SV, NA, PS, SE, or JD) for each security-related function that includes:

- Changing object authority (authorization list and objects)
- Creating, changing, and restoring user profiles
- Resetting the DST security officer password
- Changing object ownership
- Changing programs to adopt the owner's authority
- Changing system values
- Changing network attributes
- Changing subsystem routing
- Specifying a user profile name for the USER parameter on the Change Objects Description (CHGJOBD) command or the Create Job Description (CRTJOBD) command

To set up security auditing, the system requires that the QSYS/QAUDJRN journal be created before this system value is changed.

Changing the System Security Level

The QSECURITY value controls the level of security on the system. When you change the QSECURITY system value, the change does not take effect until the next IPL of the system. The security level does not take effect until after exiting the IPL Sign On display.

The possible values are:

10: Password and resource security are not active.

20: Only password security is active.

30: Password and resource security are active.

40: Password security, resource security, and operating system integrity are active. Programs that try to access objects using interfaces that are not supported fail.

Changing the Maximum Number of Sign-On Attempts

The QMAXSIGN value controls the number of sign-on attempts that are not correct by local and remote users. Incorrect sign-on attempts can be caused by a user ID that is not correct, a password that is not correct, or a user trying to sign-on a display station for which he has no authority.

When changing the QMAXSIGN value, a value (other than *NOMAX) must be enclosed in apostrophes.

The maximum number of sign-on attempts is in effect immediately after being changed. When the maximum number of sign-on attempts is reached, the QMAXSGNACN value is used to determine the action to be taken. A message is sent to the QSYSOPR message queue (and QSYSMSG message queue if it exists in library QSYS) to notify the security offifcer of a possible intrusion.

The possible values are for the QMAXSIGN value are:

15: A user can try to sign on a maximum of 15 times.

***NOMAX:** The system allows an unlimited number of invalid sign-on attempts.

Security Consideration

*NOMAX allows unlimited attempts to sign on, which may allow someone to eventually access the system. The recommended number of sign-on attempts is three. This allows three attempts to enter the correct information. Usually three attempts are enough to correct typing errors but low enough to help prevent unauthorized access.

'limit': Specify a value from 1 through 25 enclosed in apostrophes.

Changing the Action Taken When Maximum Number of Sign-On Attempts is Reached

The QMAXSGNACN value controls the action taken once the QMAXSIGN value (maximum number of sign-on attempts) is reached.

When changing the QMAXSGNACN value, the value must be enclosed in apostrophes.

The action taken for failed sign-on attempts is in effect immediately after being changed. A message is sent to the QSYSOPR message queue (and QSYSMSG message queue if it exists in library QSYS) to notify the security officer of a possible intrusion.

The possible values for the QMAXSGNACN value are:

3: Disable both the user profile and device

1: Disable the device only.

2: Disable the user profile only.

Notes:

1. The device will be disabled only if the sign-on attempts that are not valid are consecutive on the same device. One valid sign-on will reset the count of invalid sign-on attempts for the device.
2. The user profile will be disabled when the number of invalid sign-on attempts for the user reaches the value in QMAXSIGN, regardless if the invalid sign-on attempts were from the same or different devices. One valid sign-on resets the count of invalid sign-on attempts in the user profile.
3. If you have created QSYSMSG, the message sent (MSGID = CPF____) contains the user and device name in the message. Therefore, it is possible to control the disabling of the device based on the device being used.

Changing the Remote Sign-On Value

The QRMTSIGN value controls if users can bypass the sign-on display on the remote system when using the display station pass-through function or the work station function of PC Support.

The possible values are:

***FRCSIGNON:** All pass-through sessions that begin on the system must go through the normal sign-on procedure.

If the user profile names are different, the pass-through attempt will fail.

***SAMEPRF:** Pass-through sessions without going through the sign-on procedure are allowed only for users whose user profile name on the remote system is the same as the user profile name on the local system.

***VERIFY:** Pass-through sessions without going through the sign-on procedure are allowed for all pass-through requests and no checking of passwords is done if the QSECURITY value is 10.

If the QSECURITY is 20 or above, all users who attempt to pass-through automatically must use passwords. The passwords are verified before automatic sign-on occurs. If the password is not valid, the pass-through attempt is rejected.

For example, *VERIFY allows USER1 on the local system to automatically sign on the remote system as USER2 as long as USER1 specifies a valid password for USER2.

***REJECT:** Pass-through sessions are not allowed to start on the remote system.

program-name: The program specified will run at the start and end of every pass-through session. This program must meet the input and output requirements as specified in the topic "Display Station Pass-Through Program" on page 5-7 and the *Remote Work Station Guide*. This program provides a way for the security officer to tailor how sign-on requests are handled and to audit who has done pass-through to the system.

The possible library values are:

QGPL: The library QGPL is used to locate the program.

***LIBL:** The library list is used to locate the program.

***CURLIB:** The current library for the job is used to locate the program. If no current library is found in the library list, QGPL is used.

library-name: Specifies the name of the library where the pass-through program is located.

Changing the Automatic Configuration of Virtual Devices Value

The QAUTOVRT values controls the creation of virtual device descriptions on a remote system when users pass through to that system.

The system value QAUTOVRT specifies if pass-through virtual devices and TELNET full screen virtual devices (as opposed to the work station function virtual device) are automatically configured. This value can only be changed by the security officer or someone with all object (*ALLOBJ) and security administrator (*SECADM) special authority.

The possible values are:

0: No virtual devices are created.

number-of-virtual-devices: Specify a value 0 through 9999 for the maximum number of virtual devices that the user wants attached to virtual controllers that are a temporary configuration for the display station pass-through and full screen TELNET.

The *Remote Work Station Guide*, has more information about using display station pass-through. The *TCP/IP Guide* has more information about using TELNET.

Changing the Time-Out Value for Inactive Jobs

The QINACTITV value specifies the time-out interval in minutes. This provides additional security that prevents users from leaving inactive work stations signed on.

When the time-out interval is changed to a value other than *NONE, then a new inactive level for inactive jobs is started.

***NONE:** No time-out interval is specified.

interval-in-minutes: Specify a value of 5 through 300.

If a message queue is specified for the QINACTMSGQ, a user or a program can monitor the message queue for the inactive job messages and then end the job if desired. If *ENDJOB is specified, the system ends the job.

Changing the Time-Out Message Queue Value for Inactive Jobs

The QINACTMSGQ value specifies the name of the message queue to which the messages for inactive jobs are sent if QINACTITV is not *NONE. This provides additional security by preventing users from leaving inactive work stations signed on. The message queue is cleared during an initial program load (IPL). The message queue must exist before the system value can be changed to a message queue name.

Security Risk

Leaving an inactive work station signed on would allow an unauthorized person access to the system.

The possible values allowed are:

***ENDJOB:** Inactive jobs are ended. If the inactive job is a group job, all jobs associated with the group are also ended. If the job is part of a secondary job, both jobs are ended. The *ENDJOB value is equal to running the command ENDJOB JOB(name) OPTION (*IMMED) ADLINTJOBS(*ALL) against the inactive job.

message-queue-name: Messages about inactive jobs are sent to the specified message queue.

Changing the Limit Security Officer Value

The QLMTSECOFR controls whether a user with all object (* ALLOBJ) or service (*SERVICE) can sign on to any work station.

The possible values allowed are:

'1': Users with *ALLOBJ or *SERVICE authority cannot sign on to any display station unless they are specifically authorized to the display station or if user profile QSECOFR is authorized to the display station.

'0': Users with *ALLOBJ or *SERVICE authority can sign on to any display station that has the public authority specified as *CHANGE.

Changing the Password Expiration Interval

The QPWDEXPITV value controls the number of days allowed before a password must be changed. If the password is not changed in the number of days specified, the user cannot sign on until the password is changed. The system warns the user that the password is about to expire starting seven days before the expiration date.

When changing the QPWDEXPITV value, a value other than *NOMAX must be enclosed in apostrophes.

The possible values are:

***NOMAX:** The system allows an unlimited number of days that a user can use the same password.

'limit-in-days': Specify a value from 1 through 366 enclosed in apostrophes.

Changing the Display Sign-On Information Value

The QDSPSGNINF value specifies if the sign-on information display is shown.

This allows users to see the sign-on information, such as date of last sign on, and sign-on attempts that were not valid. If the password is due to expire in seven days or less, the number of days until the password expires is shown.

The possible values are:

'0': Sign-on information is not shown.

'1': Sign-on information is shown when the user signs on.

Changing the Limit Device Sessions Value

The QLMTDEVSSN value specifies if the users are limited to one device session. This value can be tailored for specific user profiles. This value does not restrict the System Request menu or a second sign-on from the same device. If users are disconnected from the system by using the Disconnect Job (DSCJOB) command, the user is allowed to sign on to the system with a new device session.

The possible values are:

'0': The system allows an unlimited number of sign-on sessions.

'1': Users are limited to one device session.

System Values That Apply to Passwords

The following system values apply to passwords. These values apply only when using the Change Password (CHGPWD) command.

Changing the Minimum Length of Passwords

The QPWDMINLEN value controls the minimum number of characters in a password. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

1: A minimum of one character is allowed for passwords.

minimum-number-of-characters: Specify a value of 1 through 10 as the minimum number of characters allowed for passwords.

Changing the Maximum Length of Passwords

The QPWDMAXLEN value controls the maximum number of characters in a password. This provides additional security by preventing users from specifying passwords that are too long and have to be recorded somewhere because they cannot be easily remembered. It can also be used to force passwords to be the same length as other systems that do not support 10-character passwords (for example, limiting passwords to 8 characters when the AS/400 system is connected to systems other than the AS/400 system). This value only applies to the Change Password (CHGPWD) command.

The possible values are:

10: A maximum of ten characters for a password are allowed.

maximum-number-of-characters: Specify a value of 1 through 10 as the maximum number of characters allowed for passwords.

Changing the Required Difference in Passwords

The QPWDRQDDIF value controls if the password must be different than the 32 previous passwords. This value provides additional security by preventing users from specifying passwords used previously. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

'0': A password can be the same as one of the previous 32 passwords.

'1': A password cannot be the same as any of the previous 32 passwords.

Changing the Restricted Characters for Passwords

The QPWDLMTCHR value limits the use of certain characters in a password. This value provides additional security by preventing users from using specific characters, such as vowels, in a password. Restricting vowels prevents users from forming actual words for their passwords. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

***NONE:** There are no restricted characters for passwords.

restricted-characters: Specify up to 10 restricted characters. The valid characters are A through Z, 0 through 9, and special characters pound (#), dollar ($), or at (@).

Changing the Restriction of Consecutive Digits in Passwords

The QPWDLMTAJC value limits the use of digits next to each other (adjacent) in a password. This value provides additional security by preventing users from using birthdays, telephone numbers, or a sequence of numbers as passwords. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

'0': Numeric characters are allowed next to each other in passwords.

'1': Numeric characters are not allowed next to each other in passwords.

Changing the Restriction of Repeated Characters in Passwords

The QPWDLMTREP value limits the use of repeating characters in a password. This value provides additional security by preventing users from specifying the same character more than once in a password. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

'0': The same characters can be used more than once in a password.

'1': The same character cannot be used more than once in a password.

Changing the Character Position Difference in Passwords

The QPWDPOSDIF value controls each position in a new password. This provides additional security by preventing users from using the same character (alphabetic or numeric) in a position corresponding to the same position in the previous password. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

'0': The same characters can be used in a position corresponding to the same position in the previous password.

'1': The same character cannot be used in a position corresponding to the same position in the previous password.

Changing the Requirement for a Numeric Character in Passwords

The QPWDRQDDGT value controls whether a numeric character is required in a new password. This value provides additional security by preventing users from using all alphabetic characters. This value only applies to the Change Password (CHGPWD) command.

The possible values are:

'0': Numeric characters are not required in new passwords.

'1': One or more numeric characters are required in new passwords

Changing the Password Approval Program

The QPWDVLDPGM value specifies the name of the password validation program. This is a program supplied by the security officer. This value allows the security officer to do additional verification of passwords. This value only applies to the Change Password (CHGPWD) command. It is recommended that this program be created into library QSYS.

The program specification includes the program name and the library name. For an example of a password approval program, see the topic "Using a Password Approval Program" on page 5-9. The possible values are:

***NONE:** No user-written program is used.

program-name: Specify a user-written program name from 1 through 10 characters.

library-name: Specifies the name of the library where the user-written program is located. If the library name is not specified, the library list (*LIBL) is used to locate the program.

Security Planning Example

The following topics in this chapter include examples of planning security for a fictitious company called STS, Incorporated. STS is a manufacturing business and has five departments:

- Accounting
- Inventory and purchasing
- Shipping and receiving
- Personnel and program support
- Manufacturing

As the security officer, you have been given the department managers' names, the department names, and a list of users and their responsibilities. It is your responsibility to set up security for these users.

Name DEPT999	Responsibility Inventory and Purchasing
J Beaker	Manager
L Greene	Fill parts requisitions for DEPT662
K Miller	Stock parts from DEPT778 (secondary programmer)
R Smith	Track inventory and place purchase orders

Name DEPT778	Responsibility Shipping and Receiving
O Lock	Manager
B Jones	Ship finished parts to customers
R Swanson	Receive parts ordered by DEPT999

Name DEPT662	Responsibility Manufacturing
P Smith	System operator and programming support
W Wendt	Manager
J Peterson	Scheduling jobs and controlling parts
S Burn	Assembly
R Stack	Assembly
M Cushion	Assembly
P Packerd	Assembly

The tasks the departments perform are controlled by application programs stored in several libraries.

In the following examples, three departments are used. The following is a list of files, libraries, and programs that STS, Incorporated uses in the business:

- The inventory and purchasing department library (INVORDLIB) contains:
  - The inventory (INVMNU) program
  - The parts requisition (PARTREQ) program, the order requisition (ORDREQ) program, and the files associated with these programs

  This library is used by departments 999, 778, and 662.

Planning Resource Security

Defining who should use a resource can help you decide who should be on an authorization list and who should be a member of a group profile.

The information you specify on the Resource Security Forms will help you determine which users will be a member of a group profile and which users will be on an authorization list. Some of the information is also used later on the User Profile Form (Part 2), Resource Security, and the Authorization List Form (Part 2), Resource Security to identify objects a user or the authorization list will have authority to use. You can find a copy of these forms in Appendix F, "Planning Forms for Security."

After you decide who will use your system, complete the following sections on the Resource Security Form to help plan which users will have authority to a resource.

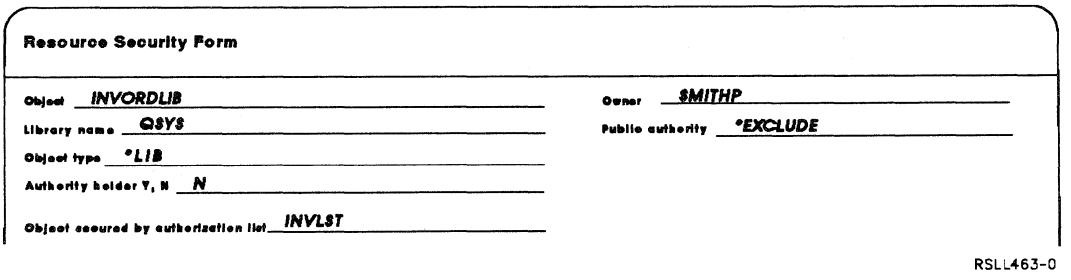


Figure 7-1. Top Part of the Resource Security Form

The top part of the form (Figure 7-1) is used to identify the resource.

Object
Specifies the name of the object.

In the example, Figure 7-1, the object is a library named INVORDLIB.

Library name
Specifies the name of the library where the object is located.

In the example, Figure 7-1, library INVORDLIB is found in library QSYS.

Object type
Specifies the type of object, such as *FILE or *PGM.

In the example, Figure 7-1, the object is a library (*LIB).

Authority holder
Specifies if this object has an authority holder. If this object has an authority holder, specify Y (Yes). The authority holder can only be used for program-described database files.

In the example, Figure 7-1, INVORDLIB does not have an authority holder.

Object secured by authorization list
Specifies the name of the authorization list if one is used to secure the object.

In the example, Figure 7-1, library INVORDLIB is secured by the authorization list INVLST.

Owner

Specify the name of the user who is the owner of the object.

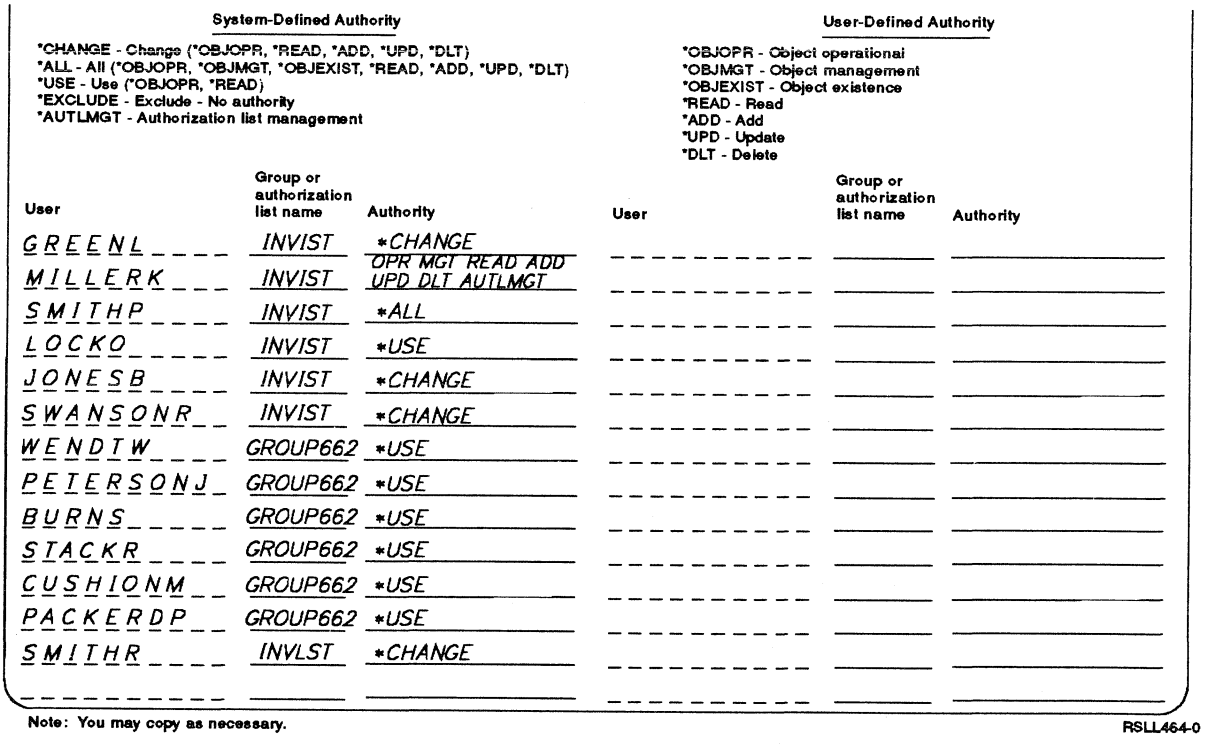
In the example, Figure 7-1, SMITHP, who is responsible for programming support, is the owner of INVORDLIB.

Public authority

Specifies the authority users will have if they do not have any specific authority to the object, who are not on the authorization list (if one is specified), whose group profile does not have any specific authority to the object, or whose group profile is not on the authorization list.

In the example, Figure 7-1 on page 7-21, users who do not have any other source of authority cannot use library INVORDLIB.

The rest of the form is used to identify the users and their authority to the object. See Figure 7-2.



System-Defined Authority

*CHANGE - Change (*OBJOPR, *READ, *ADD, *UPD, *DLT)
*ALL - All (*OBJOPR, *OBJMGT, *OBJEXIST, *READ, *ADD, *UPD, *DLT)
*USE - Use (*OBJOPR, *READ)
*EXCLUDE - Exclude - No authority
*AUTLMGT - Authorization list management

User-Defined Authority

*OBJOPR - Object operational
*OBJMGT - Object management
*OBJEXIST - Object existence
*READ - Read
*ADD - Add
*UPD - Update
*DLT - Delete

User	Group or authorization list name	Authority	User	Group or authorization list name	Authority
GREENL	INVIST	*CHANGE			
		OPR MGT READ ADD			
MILLERK	INVIST	UPD DLT AUTLMGT			
SMITHP	INVIST	*ALL			
LOCKO	INVIST	*USE			
JONESB	INVIST	*CHANGE			
SWANSONR	INVIST	*CHANGE			
WENDTW	GROUP662	*USE			
PETERSONJ	GROUP662	*USE			
BURNS	GROUP662	*USE			
STACKR	GROUP662	*USE			
CUSHIONM	GROUP662	*USE			
PACKERDP	GROUP662	*USE			
SMITHR	INVLST	*CHANGE			

Note: You may copy as necessary.

RSLL464-0

Figure 7-2. Bottom Part of the Resource Security Form

User profile name

Specifies the user profile names of the users who will have authority for the object.

In the example, Figure 7-2, users from three departments will have authority to library INVORDLIB.

Group profile or authorization list name

Specifies the name of the group profile that the user will be a member of or the name of the authorization list that the user will be assigned to.

In the example, Figure 7-2, users from department 662 have been assigned to group profile GROUP662.

Authority

Specifies the authority that each user will have for the object. If *EXCLUDE is specified, no other authorities can be specified.

Keeping in mind the security planning example, users from three departments will have different authorities to the library INVORDLIB.

The following describes the functions a user can perform for each of the authorities. You can specify *one* of the following system-defined authorities:

***CHANGE:** Change authority combines object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

***ALL:** All authority combines all the object authorities and data authorities. The user can control the object's existence, specify the security for the object, and change the object.

***USE:** Use authority combines object operational authority and read authority. Use authority allows the user to run a program that reads a file or displays the contents of a file.

***EXCLUDE:** Exclude authority prevents the user from using the object or its contents. No other authority can be specified with *EXCLUDE.

or specify *one or more* of the following user-defined object authorities and data authorities:

***OBJMGT:** Object management authority allows the user to specify the authority for the object, move or rename the object, and add members to database files.

***OBJEXIST:** Object existence authority allows the user to delete the object, free storage of the object, save and restore the object, and transfer ownership of the object.

***OBJOPR:** Object operational authority allows the user to look at the description of an object and use the object as determined by the user's data authorities to the object.

***READ:** Read authority allows the user to look at the contents of an entry in an object or to run a program that reads the object.

***ADD:** Add authority allows the user to add entries to an object; for example, add job entries to a job queue or add records to a file.

***UPD:** Update authority allows the user to change the data in an object, such as a journal, a message queue, or a data area.

***DLT:** Delete authority allows the user to remove entries from an object; for example, delete messages from a message queue or delete records from a file.

Planning User Profiles

The Resource Security Form provides information that helps determine who will be a member of a group profile and who will be on an authorization list. In the following topic, the information from Figure 7-2 on page 7-22 allows you to plan group profiles and individual user profiles.

In the following topic "Group Profile Example" on page 7-24, the users listed on the Resource Security Form with *USE authority are set up as members of a group profile.

A user can be a member of only one group profile but can be on several authorization lists. An authorization list can be created for users who are already members of a group profile or for users who need different authorities than other users on the authorization list.

Group Profile Example

In the following example of User Profile Form (Part 1), only the parameters that are used most for group profiles are discussed. The remaining parameters on the User Profile Form (Part 1) are discusses in the "Individual User Profile Example" on page 7-28.

Keeping in mind the security planning example, departments 662, 778, and 999 use the inventory program in library INVORDLIB. The inventory program is an interactive program that allows the user to enter a part name and receive the status of parts in stock, parts on order, and parts in receiving. The users in department 662 can only enter a part name. They cannot change any other field on the display. Because all members of department 662 need the same authority to the programs and files, a group profile is created to give each member the same authority.

User Profile Form (Part 1)

A general description of each item on this form was discussed in Chapter 3, "User Profiles." This topic describes the values you can specify when creating a group profile. In this example, the values that are not specified use the system-supplied defaults except the user profile name. The defaults for the parameters on the form are enclosed in parentheses. For more information about creating a group profile on the system, see the topic "Creating a Group Profile" on page 8-12.

In Figure 7-3, the top part of the form is used to identify the user, the user's position, and what the user is responsible for in his position. Since a user profile can be used as a group profile, this profile is specified as a group profile.

User Profile Form (Part 1)

Name *John Peterson*
Position *Parts analyzer*
Responsibilities *Job scheduling and parts control*

Use profile (Yes, No) *YES*
Group profile (Yes, No) *NO*
Group member (Yes, No) *YES*
Group profile name *GROUP662*

The default values are in parentheses.
Required = *

*User *PETERSONJ*
Password (*USRPRF) *X2C9A*
Set password to expire (*NO) **YES*
Profile status (*ENABLED)
User class (*USER)
Assistance Level (*SYSVAL)

Current library (*CRTDFT)
Initial program *NONE) *INVMNU*
Initial program (*NONE)
Library name (*LIBL) *INVORDLIB*
Initial menu (MAIN) **SIGNOFF*
Library name (*LIBL)
Limited capability (*NO) **YES*
Text (*BLANK) *'Inventory inquiry use'*

RV2L055-0

Figure 7-3. Top Part Of User Profile Form (Part 1)

In the following parameter descriptions, the parameter keywords are shown in parentheses after the parameter name. Some of the values listed under the parameters are underlined. These values are the system-supplied defaults.

Security-Related Parameters

User profile name (USRPRF)

Specifies the user or group name you want to assign to this user profile. The user profile name is the same as the user ID that the user types on the Sign-On display.

In the example, Figure 7-3 on page 7-24, the profile GROUP662 is the group profile for department 662.

Security Consideration

If you are communicating with other systems, a maximum of 8 characters is recommended.

Password (PASSWORD)

Specifies the passwords assigned to the user or group profile.

In the example, Figure 7-3 on page 7-24, users who become members of the group cannot sign on with the group profile name. They must sign on with their own user profile name and password. If you allow the members to sign on with the group profile name, you do not know which member of the group signed on.

***USRPRF:** The password for this user is the same as the user profile name.

***NONE:** No password is allowed with this user profile.

user-password: Specify an alphameric character string (10 characters or less) that identifies the user with his own user profile.

Text (TEXT)

Specify the description of the user profile.

In the example, Figure 7-3 on page 7-24, the user profile is identified as the group profile for department 662.

***BLANK:** No text is specified.

'description': Specify no more than 50 characters, enclosed in apostrophes.

At the bottom of the User Profile Form (Part 1), you can specify the names of the users who will be members of the group. See Figure 7-4 on page 7-26.

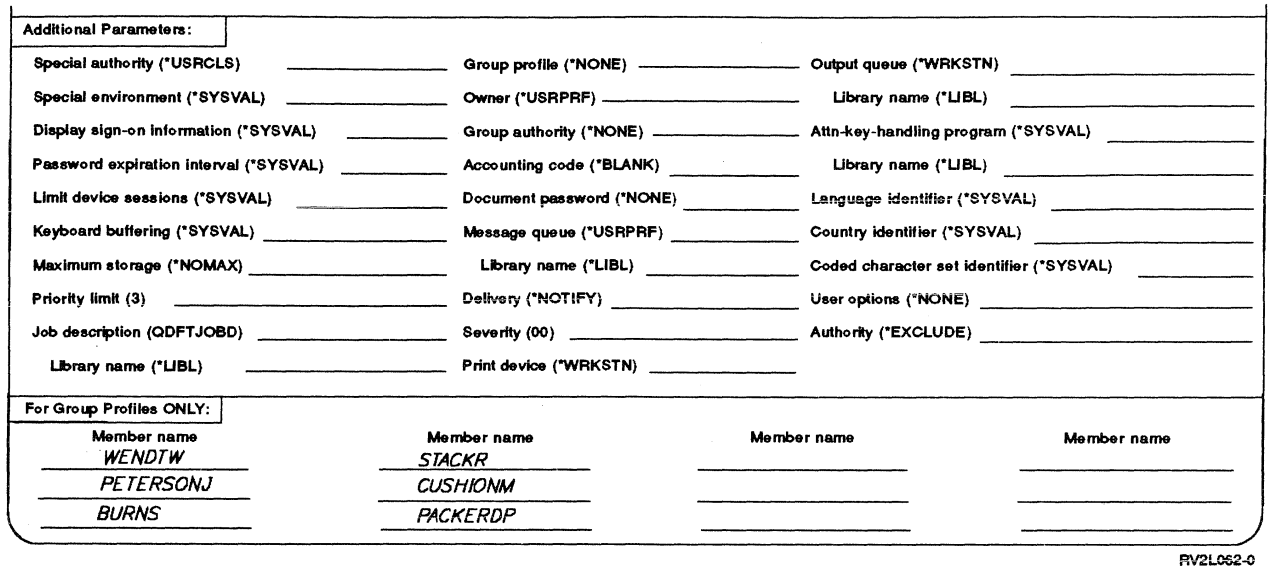


Figure 7-4. Bottom Part of User Profile Form (Part 1)

User Profile Form (Part 2), Resource Security

When you have completed User Profile Form (Part 1), you can use the User Profile Form (Part 2) to identify and plan the user's authority to specific resources on the system.

In the example, Figure 7-5 on page 7-27, the User Profile Form (Part 2), Resource Security shows a list of resources that group profile GROUP662 will use.

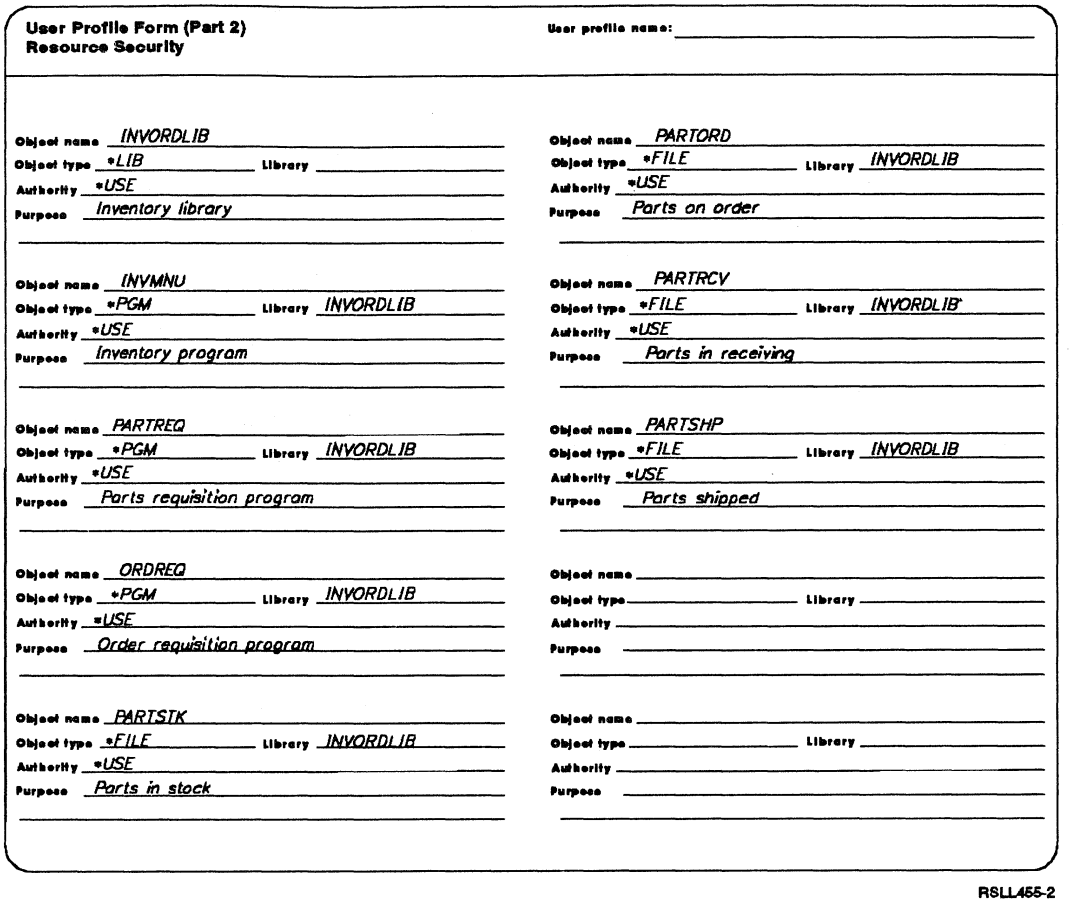


Figure 7-5. User Profile Form (Part 2) Resource Security

Object name
Specifies the name of the object the user will have authority to.

In the example, Figure 7-5, the name of the library and the names of the files and programs in the library are listed.

Object type
Specifies the type of object such as *FILE or *PGM.

In the example, Figure 7-5, files, libraries, and programs are listed.

Library
Specify the name of the library where the object is located.

Authority
Specifies the user's authority for the object. A maximum of eight authorities can be specified.

In the example, Figure 7-5, the group profile has *USE authority to all the objects listed.

You can specify *one* of the following system-defined authorities:

***CHANGE:** Change authority combines object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

***ALL:** All authority combines all the object authorities and data authorities. The user can control the object's existence, specify the security for the object, and change the object.

***USE:** Use authority combines object operational authority and read authority. Use authority allows the user to run a program that reads a file or displays the contents of a file.

***EXCLUDE:** Exclude authority prevents the user from using the object or its contents. No other authority can be specified with *EXCLUDE.

or specify *one or more* of the following user-defined object authorities and data authorities:

***OBJMGT:** Object management authority allows the user to specify the authority for the object, move or rename the object, and add members to database files.

***OBJEXIST:** Object existence authority allows the user to delete the object, free storage of the object, save and restore the object, and transfer ownership of the object.

***OBJOPR:** Object operational authority allows the user to look at the description of an object and use the object as determined by the user's data authorities to the object.

***READ:** Read authority allows the user to look at the contents of an entry in an object or to run a program that reads the object.

***ADD:** Add authority allows the user to add entries to an object; for example, add job entries to a job queue or add records to a file.

***UPD:** Update authority allows the user to change the data in an object, such as a journal, a message queue, or a data area.

***DLT:** Delete authority allows the user to remove entries from an object; for example, delete messages from a message queue or delete records from a file.

Purpose

Specify what this object is used for.

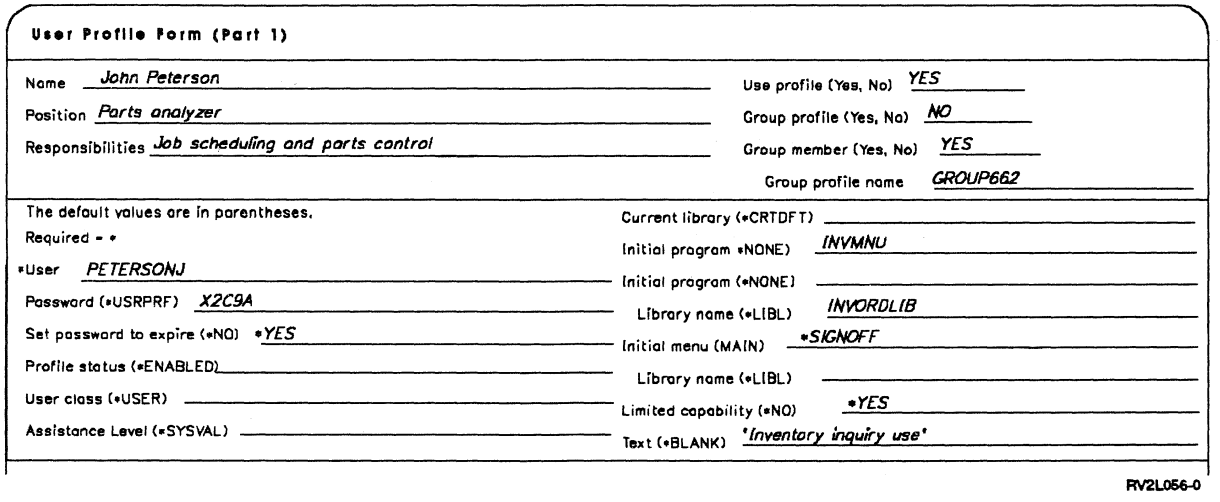
Individual User Profile Example

In the following example, a user profile for John Peterson is being planned. John is a member of department 662 and is also a member of the group profile GROUP662. PETERSONJ is limited to the inventory program. The INVMNU program is specified for his initial program to run when he signs on. When the initial program completes, PETERSONJ is signed off the system. In the example, PETERSONJ does not create any objects nor does the group. However, if PETERSONJ did create objects, you can specify that the group profile will be the owner of the objects he creates.

User Profile Form (Part 1)

The following parameters are divided into two parts. The first part contains parameters that help you control a user's access to the system and to system resources. The second part contains additional parameters that are not directly related to security but help control how a user functions on the system.

An example of the User Profile Form (Part 1) for PETERSONJ follows:


User Profile Form (Part 1)

Name John Peterson
Position Parts analyzer
Responsibilities Job scheduling and parts control
Use profile (Yes, No) YES
Group profile (Yes, No) NO
Group member (Yes, No) YES
Group profile name GROUP662

The default values are in parentheses.
Required - *
*User PETERSONJ
Password (*USRPRF) X2C9A
Set password to expire (*NO) *YES
Profile status (*ENABLED)
User class (*USER)
Assistance Level (*SYSVAL)
Current library (*CRTDFT)
Initial program *NONE) INVMNU
Initial program (*NONE)
Library name (*LIBL) INVORDLIB
Initial menu (MAIN) *SIGNOFF
Library name (*LIBL)
Limited capability (*NO) *YES
Text (*BLANK) 'Inventory inquiry use'

RV2L056-0

Figure 7-6. Top Part of User Profile Form (Part 1)

Security-Related Parameters

User profile name (USRPRF)

Specifies the user name you want to assign to this user profile. The user profile name is the same as the user ID that the user types on the sign-on display.

Security Consideration

If you are communicating with other systems, a maximum of 8 characters is recommended.

In the example, Figure 7-6, John Peterson's user profile is named PETERSONJ.

Password (PASSWORD)

Specifies the password that the user will enter to sign on.

In the example, Figure 7-6, PETERSONJ is assigned X2C9A as his initial password. You can specify that PETERSONJ must change his initial password when he signs on by specifying *YES for the set password to expired (PWDEXP) parameter.

***USRPRF:** The password for this user is the same as the user profile name.

***NONE:** No password is used by this user.

user-password: Specifies an alphameric character string (10 characters or less) that identifies the user with his own user profile.

Set password to expired (PWDEXP)
Specifies that the password should be set to expired for the user. If the password is set to expired, the user will be required to change the password during the next sign-on.

In the example, Figure 7-6 on page 7-29, PETERSONJ must change his password during sign-on. The Sign-on Information display will be shown. From this display, he can press F9 (Change password).

***NO:** The password is not set to expired.

***YES:** The password is set to expired.

Status (STATUS)
Specifies profile status.

***ENABLED:** The profile created is valid for sign-on.

***DISABLED:** The profile created is not valid for sign-on until an authorized user enables it again.

User class (USRCLS)
Specifies the class of user. Each user class has a specific set of special authorities associated with it. (USRCLS)

In the example, Figure 7-7 on page 7-33, PETERSONJ is specified as *USER and his special authorities are determined by that user class.

***USER:** If the system security level is 30 or above, no special authority is granted.

If the system security level is 10 or 20, the user is granted all object authority and save system authority.

***SECOFR:** At all system security levels, the user is granted all special authorities: all object authority, security administrator authority, save system authority, job control authority, service authority, and spool control authority.

***SECADM:** If the system security level is 30 or above, the user is granted security administrator authority, save system authority, and job control authority.

If the system security level is 10 or 20, the user is granted all object authority, security administrator authority, save system authority, and job control authority.

***PGMR:** If the system security level is 30 or above, the user is granted save system authority and job control authority.

If the system security level is 10 or 20, the user is granted all object authority, save system authority, and job control authority.

***SYSOPR:** If the system security level is 30 or above, the user is granted save system authority and job control authority.

If the system security level is 10 or 20, the user is granted all object authority, save system authority, and job control authority.

Assistance level (ASTLVL)
Specifies which user interface to use.

***SYSVAL:** The assistance level defined for the system is used.

***BASIC:** The Operational Assistant user interface is used.

***INTERMED:** The system interface is used.

***ADVANCED:** The expert system interface is used. To allow for more list entries, the option keys and the function keys are not displayed. If a command does not have an advanced (*ADVANCED) level, the intermediate (*INTERMED) level is used.

Current library (CURLIB)
Specifies the name of this user's current library.

If *YES or *PARTIAL is specified in the limited capability (LMTCPB) parameter, the user cannot change his current library value.

In the example, Figure 7-6 on page 7-29, if PETERSONJ creates objects specifying *CURLIB on a create command, the library QGPL is used as his default current library.

***CRTDFT:** This user has no current library. If objects are created into the current library using *CURLIB on a create command, the library QGPL is used as the default current library.

current-library-name: Specify the 10-character name of the library that is this user's current library after he signs on.

Initial program (INLPGM)
Specifies the default initial program for this user when he signs on.

If *YES or *PARTIAL is specified in the limited capability (LMTCPB) parameter, the user cannot change the program value.

In the example, Figure 7-6 on page 7-29, PETERSONJ is limited to the inventory menu program (INVMNU) in library INVORDLIB.

***NONE:** No program is called when the user signs on. If a menu name is specified on the initial menu (INLMNU) parameter, that menu is displayed.

program-name: Specify the name of the program that is called when the user signs on.

Library
Specifies the name of the library where the program is located.

The possible library values are:

***LIBL:** The library list that is used to locate the program.

***CURLIB:** The current library for the job is used to locate the program. If no current library entry exists in the library list, QGPL is used.

library-name: Specify the library where the program is located.

Initial menu (INLMNU)
Specifies the name of the default initial menu for the user when he signs on.

If *YES is specified for the limited capability parameter, the user cannot change the menu value.

In the example, Figure 7-6 on page 7-29, PETERSONJ is automatically signed off when the INVMNU program completes. PETERSONJ is not allowed to do any function other than functions limited to the inventory program.

MAIN: The AS/400 system Main Menu is shown.

***SIGNOFF:** The system signs off the user when the initial program completes. This is intended for users you want to limit to running the program only.

menu-name: Specify the name of the menu that is called when the user signs on.

Library

Fill in the name of the library where the menu program is located.

The possible library values are:

***LIBL:** The library list is used to locate the menu.

***CURLIB:** The current library for the job is used to locate the menu. If no current library entry exists in the library list, QGPL is used.

library-name: Specify the library where the menu is located.

Limited capability (LMTCPB)

Specifies whether the user can change the initial program, the initial menu, the current library, and the Attention-key-handling program values.

In the example, Figure 7-6 on page 7-29, PETERSONJ cannot change his initial program, initial menu, Attention-key-handling program, or his current library when he signs on.

***NO:** The initial program, initial menu, and current library values *can* be changed when the user signs on the system. A user *can* change the initial program, initial menu, current library, and Attention-key-handling program values in his own user profile with the Change Profile (CHGPRF) command.

***PARTIAL:** The initial program and current library values *cannot* be changed on the Sign On display. The initial menu value *can* be changed and commands can be run from the command line of a menu. A user *can* change the initial menu value with the Change Profile command. The initial program, current library, and the Attention-key-handling program values *cannot* be changed using the Change Profile command.

***YES:** The initial program, initial menu, and current library values *cannot* be changed on the Sign-On display. Some commands *can* be run from the command line of a menu. The commands allowed are SIGNOFF, SNDMSG, DSPMSG, DSPJOB, and DSPJOBLOG. The user *cannot* change the initial program, initial menu, current library, or the Attention-key-handling program value using the Change Profile command.

Text (TEXT)

Specifies a description of the user profile.

In the example, Figure 7-6 on page 7-29, the PETERSONJ user profile is used for inventory inquiry.

***BLANK:** No text is specified.

'description': Specify no more than 50 characters, enclosed in apostrophes.

Additional Parameters

The following parameters are not directly related to security but tailor the way a user functions on the system.

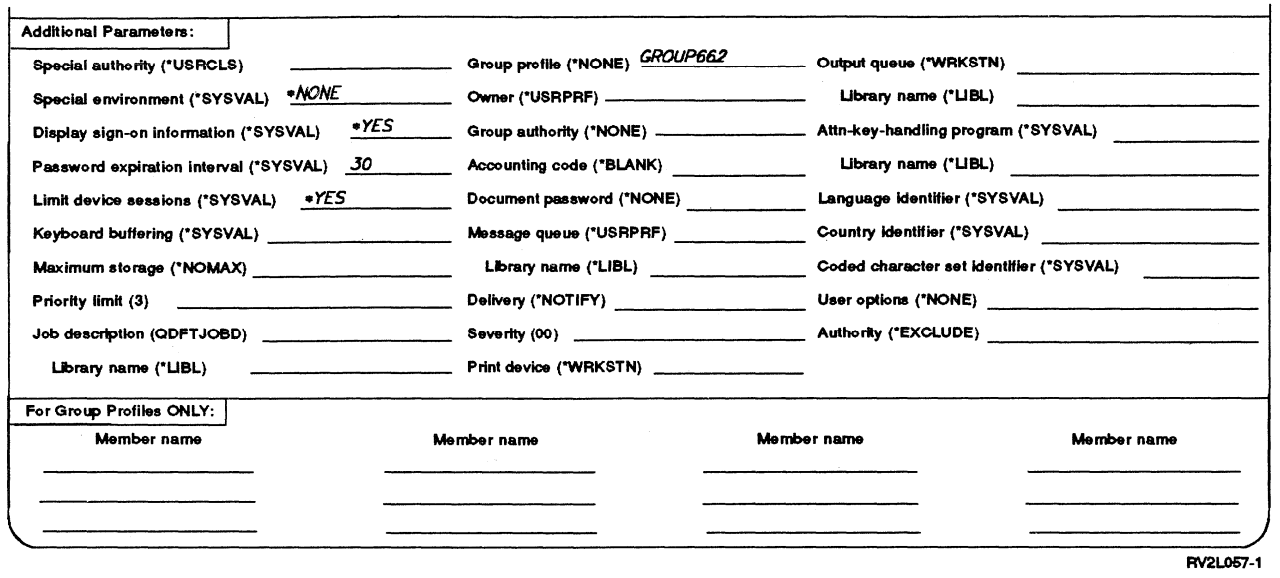


Figure 7-7. User Profile Form (Part 1)

Special authority (SPCAUT)

Specifies the special authority given to this user profile. (SPCAUT)

In the example, Figure 7-7, the special authorities for PETERSONJ are determined by the user class.

If you do not want to use the default special authorities associated with the user class, you can specify your own combination in this parameter. If you do specify values in this parameter, they override the special authorities associated with the user class, which is specified in the USRCLS parameter.

Note: The user profile that is creating or changing another user profile must have all the special authorities being granted to the other user profile.

Specify *one* of the following values:

***USRCLS:** Special authorities are granted to this user based on the value specified in the USRCLS parameter.

***NONE:** No special authority is granted to this user.

or specify *one or more* of the following values:

***ALLOBJ:** All object authority is granted to this user. The user can access any system resource regardless of whether or not any private authority exists for the user.

Security Risk

Giving a user all object special authority is a **security risk** because it allows the user to access all system resources.

***SAVSYS:** Save system authority is granted to this user. This user is given the authority to save, restore, and free storage for all objects on the system, regardless of whether or not he has object existence authority.

***JOBCTL:** Job control authority is granted to this user. The user is given the authority to change, display, hold, release, cancel, and clear all jobs that are running on the system or that are on a create job queue or output queue command that has OPRCTL(*YES) specified. The user also has the authority to perform an initial program load (IPL) of the system to start writers and to stop active subsystems.

***SECADM:** Security administrator authority is granted to this user. The user can create or change user profiles if he has authority to the Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command. This value does not allow granting special authorities that this user profile does not have.

***SERVICE:** Service authority is granted to this user. The user can perform alter service functions.

***SPLCTL:** Spool control authority is granted to this user. The user can perform all spool functions such as cancel, delete, hold, display, and release spooled files.

Special environment (SPCENV)

Specifies which environment the user will operate in.

In the example, Figure 7-7 on page 7-33, PETERSONJ will operate in no special environment.

***SYSVAL:** The system value QSPCENV is used to determine the system environment for the user after signing on the system.

***NONE:** No special environment is defined for the AS/400 system.

***S36:** The user operates in the System/36 environment after signing on the system.

Display sign-on information (DSPSGNINF)

Specifies whether the Sign-on Information display is shown when the user signs on. This allows users to see the sign-on information, such as the date of the last sign-on and sign-on attempts that were not valid. If the password is due to expire in seven days or less, the number of days until the password expires is shown.

In the example, Figure 7-7 on page 7-33, PETERSONJ sees his sign-on information when he signs on.

***SYSVAL:** The system value QDSPSGNINF is used to determine if the sign-on information is shown when the user signs on.

***NO:** The Sign-on Information display is not shown when the user signs on.

***YES:** The Sign-on Information display is shown when the user signs on.

Password expiration interval (PWDEXPITV)

Specifies the number of days before a password (from the date last changed) will expire.

In the example, Figure 7-7 on page 7-33, the password for PETERSONJ must be changed every thirty days.

***SYSVAL:** The system value QPWDEXPITV is used to determine the password expiration interval.

***NOMAX:** There is no password expiration interval.

password-expiration-interval: Specify the number of days in which the password will expire. Valid values are 1 through 366.

Limit device sessions (LMTDEVSSN)
Specifies if the number of device sessions allowed for the user should be limited to one. This does not limit the System Request menu and a second sign-on.

In the example, Figure 7-7 on page 7-33, PETERSONJ is limited to one device session.

***SYSVAL:** The system value QLMTDEVSSN is used to determine if the user is limited to one device session.

***NO:** The user is not limited to one device session.

***YES:** The user is limited to one device session.

Keyboard buffer (KBDBUF)
Specifies the keyboard buffering value used when a job is initialized for this user profile.

***SYSVAL:** The system value, QKBDBUF, is used to determine the keyboard buffering value for this profile.

***NO:** The type-ahead feature and attention-key buffering option are not active for this user profile.

***TYPEAHEAD:** The type-ahead feature is active for this user profile.

***YES:** The type-ahead feature and attention-key buffering option are active for this user profile.

Maximum storage (MAXSTG)
Specify one of the following for the maximum storage this user is allowed.

In the example, Figure 7-7 on page 7-33, PETERSONJ can be assigned as much storage as required.

***NOMAX:** As much storage as required can be assigned to this profile.

maximum-K-bytes: Specify the maximum amount of storage in kilobytes (K) (1 K equals 1024 bytes) that can be assigned to this user profile.

Priority limit (PTYLMT)
Specifies the highest scheduling priority this user is allowed to have.

In the example, Figure 7-7 on page 7-33, the default is used.

3: The user named in this profile can have a priority value no higher than 3 for scheduling any of his jobs on the system. All jobs having this priority value run before all jobs having values of 4 through 9 and after all jobs having values of 1 and 2.

priority-limit: Specify a value, 1 through 9, for the highest scheduling priority that the user is allowed. The highest priority is 1; the lowest priority is 9.

Job description (JOBD)
Specifies the job description for this user.

In the example, Figure 7-7 on page 7-33, the default is used.

QDFTJOBD: The system-supplied job description found in library QGPL is used.

job-description-name: Specify the job description name that is used to control jobs run by this user. The job parameter values specified by the user for a job use this job description if the job is specified as (JOBD(*USRPRF)).

Library

Specifies the name of the library where the job description for the user is located.

The possible library values are:

QGPL: The library QGPL is used to locate the job description QDFTJOBD.

***LIBL:** The library list is used to locate the job description.

***CURLIB:** The current library for the job is used to locate the job description. If no current library entry exists in the library list, QGPL is used.

library-name: Specify the library where the job description is located.

Group profile (GRPPRF)

Specifies the name of the group profile for the user profile. If the user profile is a group profile, this parameter cannot be specified.

In the example, Figure 7-7 on page 7-33, PETERSONJ belongs to group profile GROUP662.

Notes:

1. The current user of the Create User Profile (CRTUSRPRF) command must have *OBJMGT, *OBJOPR, *READ, *ADD, and *DLT authorities to the user profile he specifies on the GRPPRF parameter.
2. When a group profile is specified, the member whose profile is being created or changed is automatically granted *OBJOPR, *READ, *ADD, and *DLT authorities to the group profile.

***NONE:** No group profile is used with this user profile.

user-profile-name: Specify the name of a group profile of which this user profile is a member. The authority of the group profile determines this member's authority to objects owned by the group profile.

Owner (OWNER)

Specifies which user profile is the owner of objects created by this user profile.

In the example, Figure 7-7 on page 7-33, the PETERSONJ user profile owns objects (if any) that PETERSONJ creates as a member of the group.

***USRPRF:** The user profile used with the job is made the owner of the object.

***GRPPRF:** The group profile is made the owner of newly created objects and is given all (*ALL) authority to the object. The user profile used with the job does not have any specific authority to the object. If *GRPPRF is specified, you must specify a group profile name in the GRPPRF parameter, and the GRPAUT parameter must be *NONE.

Group authority (GRPAUT)

Specifies the authority given to the group profile for the objects created by this user profile.

In the example, Figure 7-7 on page 7-33, other members of GROUP662 have *USE authority to any objects that PETERSONJ creates as a member of the group.

If *GRPPRF is specified on the OWNER parameter, this parameter must be *NONE.

***NONE:** No authority is given to the group profile when this user creates objects.

***ALL:** All authority combines all the object authorities and data authorities. The user can control the object's existence, specify the security for the object, and change the object.

***CHANGE:** Change authority combines object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

***USE:** Use authority combines object operational authority and read authority. The user can run a program or display the contents of a file.

***EXCLUDE:** Exclude authority prevents the user from using the object or its contents.

Accounting code (ACGCDE)

Specifies the accounting code used by this user.

In the example, Figure 7-7 on page 7-33, the system-supplied default is used.

***BLANK:** An accounting code of 15 blanks is assigned to this user profile.

accounting-code: Specify the 15-character accounting code used by jobs that get their accounting code from this user profile. If less than 15 characters are specified, the string is padded with blanks on the right.

Document password (DOCPWD)

Specifies the document password used by this user. The *Using OfficeVision/400* Word Processing* contains additional information about document passwords.

In the example, Figure 7-7 on page 7-33, no document password is specified for PETERSONJ.

***NONE:** No document password is used by this user.

document-password: Specify a document password for this user. The password must consist of from 1 through 8 characters (letters A through Z and numbers 0 through 9). The first character of the document password must be alphabetic; the remaining characters can be alphameric. Embedded blanks, leading blanks, and special characters are not allowed.

Message queue (MSGQ)

Specifies the message queue to be used by this user. A message queue will be created by the system if one is not specified.

In the example, Figure 7-7 on page 7-33, a message queue with the same name as the user profile name is created for PETERSONJ.

***USRPRF:** A message queue with the same name as that specified in the USRPRF parameter is used as the message queue for this user. This message queue is located in library QUSRSYS.

message-queue-name: Specify the message queue name that is used for this user.

Library

Specifies the name of the library where the message queue is located.

The possible library values are:

***LIBL:** The library list is used to locate the message queue.

***CURLIB:** The current library for the job is used to locate the message queue. If no current library entry exists in the library list, QGPL is used.

library-name: Specify the library where the message queue is located.

Delivery (DLVRY)

Specifies the type of delivery for messages sent to this user.

In the example, Figure 7-7 on page 7-33, PETERSONJ is notified when a message arrives in his message queue.

***NOTIFY:** The job that the message queue is assigned to is notified when a message arrives at the message queue. For interactive jobs at a work station, the audible alarm is sounded and the message-waiting light is turned on. The type of delivery cannot be changed to *NOTIFY if the message queue is also being used by another user.

***BREAK:** The job that the message queue is assigned to is interrupted when a message arrives at the message queue. If the job is an interactive job, the audible alarm is sounded (if the alarm is installed). The type of delivery cannot be changed to *BREAK if the message queue is also being used by another user.

***HOLD:** The messages are held in the message queue until they are requested by the user or program.

***DFT:** Messages requiring replies are answered with their default reply; information-only messages are ignored.

Severity code (SEV)

Specifies the severity level used for messages sent to this user.

Note: If *BREAK or *NOTIFY is specified on the DLVRY parameter and is in effect when a message arrives at the queue, the message is delivered if the severity code associated with the message is equal to or greater than the value specified here. Otherwise, the message is held in the queue until it is requested.

In the example, Figure 7-7 on page 7-33, the default is used.

00: If a severity code is not specified, 00 is used.

severity-code: Specify a value, 00 through 99, for the lowest severity code that a message can have and still be delivered if the message queue is in break or notify delivery mode. Any 2-digit value can be specified, even if no severity code has been defined for it (either defined by the system or by the user).

Print device (PRTDEV)
Specifies the printer used by this user.

In the example, Figure 7-7 on page 7-33, the printer device specified for the system value is used for PETERSONJ.

***WRKSTN:** The output queue assigned to the user's work station is used.

***SYSVAL:** The default system printer specified in the system value QPRTDEV is used.

printer-device-name: Specify the name of a printer that is used to print the output for this user.

Output queue (OUTQ)
Specifies the name of the output queue used by this user. The output queue must already exist.

In the example, Figure 7-7 on page 7-33, an output queue with the same name as the printer device specified for the system value (QPRTDEV) is the output queue for PETERSONJ.

***WRKSTN:** The output queue assigned to the user's work station is used.

***DEV:** An output queue with the same name as specified on the PRTDEV parameter is used for this user.

output-queue-name: Specify the name of the output queue that is used for this user.

Library
Specifies the name of the library where the output queue is located.

The possible library values are:

***LIBL:** The library list is used to locate the output queue.

***CURLIB:** The current library for the job is used to locate the output queue. If no current library entry exists in the library list, QGPL is used.

library-name: Specify the library where the output queue is located.

Attention-key-handling program (ATNPGM)
Specifies the Attention-key-handling program for this user.

If *YES or *PARTIAL is specified on the limited capability parameter, the Attention-key-handling program value cannot be changed with the Change Profile (CHGPRF) command.

In the example, Figure 7-7 on page 7-33, PETERSONJ does not have an Attention-key-handling program.

***SYSVAL:** The system value QATNPGM is used.

***NONE:** No Attention-key-handling program is used by this user.

***ASSIST:** QEZMAIN is used.

program-name: Specify the name of the program that is used by this user.

Library
Specifies the name of the library where the Attention-key-handling program is located.

The possible library values are:

***LIBL:** The library list is used to locate the Attention-key-handling program.

***CURLIB:** The current library for the job is used to locate the Attention-key-handling program. If no current library entry exists in the library list, QGPL is used.

library-name: Specify the library where the Attention-key-handling program is located.

Language identifier (LANGID) (For Version 2 Release 1.1)
Specifies the language identifier to be used by the system for this user.

***SYSVAL:** The system value QLANGID is used to determine the language identifier.

language identifier: Specify the language identifier for this user.

Country identifier (CNTRYID) (For Version 2 Release 1.1)
Specifies the country identifier to be used by the system for this user.

***SYSVAL:** The system value QCNTRYID is used to determine the country identifier.

country identifier: Specify the country identifier for this user.

Coded character set identifier (CCSID) (For Version 2 Release 1.1)
Specifies the coded character set identifier to be used by the system for this user.

***SYSVAL:** The system value QCCSID is used to determine the coded character set identifier.

coded character set identifier: Specify the coded character set identifier for this user.

User options (USROPT)
Specifies the level of detail the user will see, how the Page Up and Page Down keys will function, and if status messages will be shown to the user.

In the example, Figure 7-7 on page 7-33, PETERSONJ is limited to his initial program. He does not need detailed information shown.

When a value other than *NONE is specified for this parameter, the system presents detailed information without any action by the experienced user.

***NONE:** No detailed information is shown to the user.

***CLKWD:** Keywords are shown instead of the possible parameter values when a control language (CL) command is displayed.

***EXPERT:** More detailed information is initially shown when the user is performing display and edit object operations to define or change the system (such as edit or display object authority).

***HLPFULL:** A full screen is used to show the user help information.

***PRTMSG:** A message is sent to the message queue of this user when a spooled file is printed for this user.

***ROLLKEY:** The actions of the Page Up and Page Down keys are reversed.

***NOSTSMSG:** Status messages usually shown at the bottom of the display are not shown to the user.

***STSMSG:** Status messages are displayed when sent to the user.

Authority (AUT)

Specifies the authority the public will have to this user profile.

In the example, Figure 7-7 on page 7-33, no other user can use the PETERSONJ user profile.

***EXCLUDE:** Exclude authority prevents the user from using the object or its contents. No other authority can be specified with *EXCLUDE.

***CHANGE:** Change authority combines object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

***ALL:** All authority combines all object authorities and data authorities. The user can control the object's existence, specify the security for the object, and change the object.

***USE:** Use authority combines object operational authority and read authority. Use authority allows the user to run a program that reads a file or display the contents of a file.

Planning an Authorization List

The Authorization List Form is divided into two parts. Part 1 is for planning the authorization list. Part 2 is for planning the resources that will be secured by the authorization list.

Authorization List Form (Part 1)

The information at the top of the form is used to identify the authorization list, the owner, and the public authority for the authorization list. This information is used when your create the authorization list. For more information about creating an authorization list, see the topic "Creating an Authorization List" on page 8-31.

The rest of the form is used to specify who is on the list and what authority each user has for objects. For more information about creating an authorization list, see "Creating an Authorization List" on page 8-31.

Security Consideration

Authorization lists cannot be used to secure user profiles or other authorization lists. You can use one authorization list to secure multiple objects, but an object can be secured by only one authorization list.

Authorization List Example

Keeping in mind the resource planning example, departments other than 662 use the inventory library. You can create an authorization list and add users and their authority to the list. The Resource Security Form allowed you to identify users that could be on an authorization list. In the following example, departments 778 and 999 are used. The following is an example of how the top part of the authorization list may appear on the Authorization List.

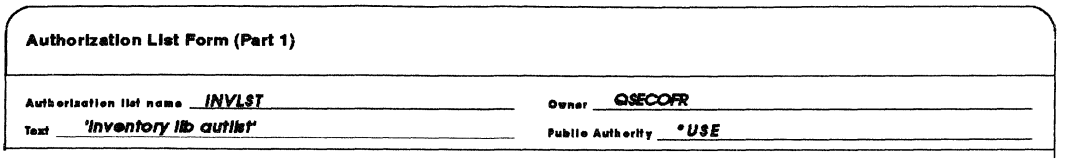


Figure 7-8. Top Part of Authorization List Form (Part 1)

Authorization list name (AUTL)
Specifies the name of the authorization list you are creating. A maximum of 10 alphameric characters can be used.

In the example, Figure 7-8, the name INVLST is used to identify it as the inventory library authorization list.

Text (TEXT)
Specifies a description of the authorization list.

In the example, Figure 7-8, the list is identified as the inventory library authorization list.

***BLANK** No text is specified.

'description': Specify no more than 50 characters, enclosed in apostrophes.

Owner
Specifies who the owner of the authorization list will be. The user who creates the authorization list is the owner. Ownership can be transferred to another user with the Change Object Owner (CHGOBJOWN) command.

In the example, Figure 7-8, the security officer (QSECOFR) user profile is the owner of the authorization list.

Public authority (AUT)
Specifies the authority that users are given if they do not have any authority specifically given to them for the authorization list, or for objects secured by the authorization list. The public authority on the authorization list is used only if the objects secured by the authorization list have their public authority specified as *AUTL.

In the example, Figure 7-8, the public has *USE authority.

***CHANGE:** Change authority combines object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

***ALL:** All authority combines all object authorities and data authorities. The user can control the object's existence, specify the security for the object, and change the object.

***USE:** Use authority combines object operational authority and read authority. Use authority allows the user to run a program that reads a file or display the contents of a file.

***EXCLUDE:** Exclude authority prevents the user from using the object or its contents. No other authority can be specified with *EXCLUDE.

Adding Users to an Authorization List

The bottom part of the Authorization List Form (Part 1) is used to plan users on the authorization list. An entry on the authorization list consists of a user profile name and the authorities associated with that user on the authorization list.

The following is an example of how the bottom part of the authorization list may appear.

Authorization List Form (Part 1)

Authorization list name: INVLST Owner: QSECOFR

Text: 'Inventory lib autlist' Public Authority: *USE

User	Object Authority *ALL *CHANGE *USE *EXCLUDE	Auth List Mgt	Object: Opr	Object: Mgt	Object: Exist	Data: Read	Data: Add	Data: Update	Data: Delete
SMITHP	*ALL	*MGT							
GREENL	*CHANGE								
MILLERK			X	X		X	X	X	X
JONESB	*CHANGE								
SMITHR	*CHANGE								
LOCKO	*USE								

RSLL462-3

Figure 7-9. Bottom Part of Authorization List Form (Part 1)

Consider the following when adding users to an authorization list:

- To add users to an authorization list, you must be: the owner of the authorization list, a user with authorization list management authority on the authorization list, or a user with all object (*ALLOBJ) special authority.
- Authorization list management authority allows a user to manage the authorization list and, therefore, to manage the authorities for all the objects the list secures.

  A user with authorization list management authority can add users and give the users only the same specific authorities that he has.
- When an authorization list is created, only the owner of the list or a user with all object (*ALLOBJ) special authority can add a user with authorization list management authority.

User (USER)

Specifies the name of users to be added to the authorization list.

In the example, Figure 7-9, members of three departments are used.

Authority (AUT)

Specify the authority you are giving the specified users for objects secured by the authorization list. A maximum of eight authorities can be specified.

Specify *one* of the following in the *Object Authority* column:

***CHANGE:** Change authority combines object operational authority and all the data authorities. The user can add, change, and delete entries in an object, or read the contents of an entry in the object.

***ALL:** All authority allows the user to perform all operations on the object except those limited to the owner or controlled by authori-

zation list management authority. The user can control the object's existence, specify the security for the object, and change the object.

***USE:** Use authority allows the user to run a program that reads a file or displays the contents of a file. Use authority combines object operational authority and read authority.

***EXCLUDE:** Exclude authority prevents the user from using the object or its contents. No other authority can be specified with *EXCLUDE except *AUTLMGT.

or specify *one or more* of the following object authorities and data authorities in the *Object* or *Data* column.

***AUTLMGT:** Authorization list management authority allows the user to add users to, and remove users from, the authorization list and change users' authorities on the authorization list. The user cannot add or remove an authority for a user if he does not have that authority.

***OBJMGT:** Object management authority allows the user to specify the authority for the object, move or rename the object, and add members to database files.

***OBJEXIST:** Object existence authority allows the user to delete the object, free storage of the object, save and restore the object, and transfer ownership of the object.

***OBJOPR:** Object operational authority allows the user to look at the description of an object and use the object as determined by the user's data authorities to the object.

***READ:** Read authority allows the user to look at the contents of an entry in an object or to run a program that reads the object.

***ADD:** Add authority allows the user to add entries to an object; for example, add job entries to a job queue or add records to a file.

***UPD:** Update authority allows the user to change the data in an object, such as a journal, a message queue, or a data area.

***DLT:** Delete authority allows the user to remove entries from an object; for example, delete messages from a message queue or delete records from a file.

Authorization List Form (Part 2), Resource Security

When you have completed the Authorization List Form (Part 1), you can use the Authorization List Form (Part 2) to identify and plan the resources to be secured by the authorization list.

The Authorization List Form (Part 2) is designed to help you specify an authorization list for an object with Grant Object Authority (GRTOBJAUT) command or the Edit Object Authority (EDTOBJAUT) command.

An example of the Authorization List Form (Part 2) follows.

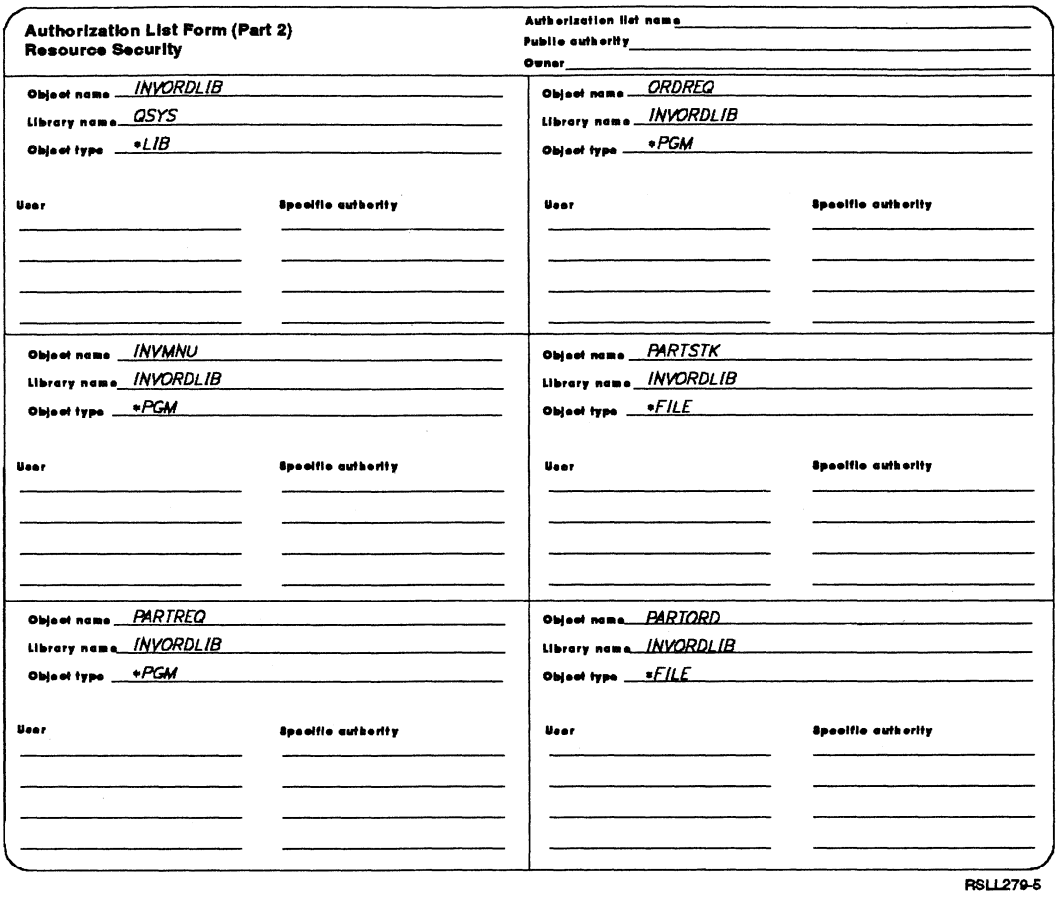


Figure 7-10. Authorization List Form (Part 2) Resource Security

Object name
Specifies the name of the object whose authority will come from the authorization list.

In the example, Figure 7-10, the name of the library and the names of the files and programs are listed.

Library name
Specifies the name of the library where the object is located.

Object type
Specifies the type of object, such as *FILE or *PGM.

In the example, Figure 7-10, the object types are files, libraries, and programs.

Specific authority
This item is used to identify a user of the object whose authority for the specific object does not come from the authorization list. This user has been given specific authority for the object.

In the example, Figure 7-10, no users have been given authority specifically for the objects. Users on the authorization list use the authority specified for them on the authorization list.

When you have completed planning the users' authorities for objects, authorization lists, and user profiles, you can use the information to set up the group

profiles, authorization lists, and user profiles on the system. When all users are set up on the system and have the authorities they need, you can change the security level to 30 (if you have not already done so) to activate resource security.

Security Officer's Checklist

An organization can do these items from the following list that meet their security requirements.

Physical Security

___ The display station designated as the console is restricted.
___ Offline media is saved and protected from damage and theft.
___ Security officer sign-on is limited to specific devices.

User Controls

___ The IBM-supplied profiles passwords are changed.
___ The IBM passwords for dedicated service tools (DST) are changed.
___ A password change is required every 30 to 90 days.
___ Trivial passwords are prevented by using the system values to set the password rules and by using password approval programs.
___ Users do not share a common password.
___ Users that are limited to menus have LMTCPB(*YES) specified to limit use of commands from system menus and to prevent users from specifying an initial program, initial menu, or current library on the sign-on display.
___ Employees are removed from the system immediately when they are transferred or released.
___ Programmers are restricted from production libraries.
___ Owners of objects annually verify the authorized users including user *PUBLIC access.
___ Management annually verifies the users authorized to the system.
___ Management quarterly verifies the users with *ALLOBJ special authority.

User Profiles

___ Each user is assigned a unique user profile.
___ Users can change their own passwords. Allowing users to define their own password reduces the need for users to write down their passwords.
___ User profiles with *ALLOBJ special authority are limited, and are not used as group profiles.
___ Critical user profiles are verified when using the program adopt function to make sure they are accessing authorized functions.

Authorization Control

___ Owners of data understand their obligation to authorize users on a need-to-know basis.
___ Sensitive data is not public.
___ Job descriptions with a public authority are specified as USER(*RQD).
___ Job descriptions that specify a user profile name have the public authority specified as *EXCLUDE.
___ User profiles that submit jobs using a job description containing a user profile name must have *USE authority to the user profile.
___ User profiles sign on by pressing the Enter key on the Sign On display. A work station entry for the user references a job description that has a user profile name specified for the USER parameter.

___ Control the library list in application programs to prevent a library that contains a similar program from being added before the production libraries.
___ Programs that adopt are used only when required.
    ___ Verify any programs that adopt the authority of critical user profiles by using the Display Program Adopt (DSPPGMADP) command.
    ___ Inspect adopting programs to prevent the user of the program from excess function; for example, command entry while running under the adopted profile.
    ___ Verify that programs adopt the minimum authority level needed. For example, the owner's profile for the object is adopted rather than the QSECOFR profile.

Unauthorized Access

___ System value QMAXSIGN limits the number of access attempts.
___ Message queue QSYSMSG is created and monitored.
___ History log messages in the CPF2200 range that report authorization failures are reviewed for repeated attempts of a user.
___ Programs fail that attempt to access objects using interfaces that are not supported.
___ Security-related events are logged to the security auditing journal (QAUDJRN).
___ User ID and password are required

Communications

___ Telephone communications is protected by call-back procedures.
___ Encryption is used on sensitive data.
___ All communications jobs require user ID and password.

Chapter 8. Setting Up Security

Chapter 6, "Auditing Security for the AS/400 System" provides information to help you decide which users will have authority to specific objects and if the users will be on an authorization list or a member of a group profile. If you have finished planning, you can use the completed planning forms to enter the information on the system.

Descriptions of the parameters shown on the following displays are found in Chapter 7, "Security Recommendations and Planning" of this manual. For parameters that you do not specify on the displays or on the commands, the system-supplied defaults are used.

The system security level must be at 30 or above for your security measures to be effective. The following topics provide step-by-step tasks that the security officer can perform to establish security on the system at security level 30 or above.

- Changing the QSECOFR user profile password
- Resetting the DST passwords to the system-supplied defaults
- Changing the DST passwords
- Changing the operating system install security
- Changing the system values that affect security
- Working with group and user profiles
- Working with authorization lists

Changing the IBM-Supplied User Profile Passwords

The following shows how to change the security officer's password from the default password shipped with the system. It is important that the security officer change this password as soon as the initial program load (IPL) of the system is complete.

Other IBM-supplied user profiles should also have their default passwords changed. These profiles are: QPGMR, QSYSOPR, QSRV, QSRVBAS, and QUSER.

1. Sign on to the system by entering QSECOFR for the *User* prompt and QSECOFR for the *Password* prompt, and press the Enter key. When you enter any password (for example, QSECOFR), it is not shown on the display.

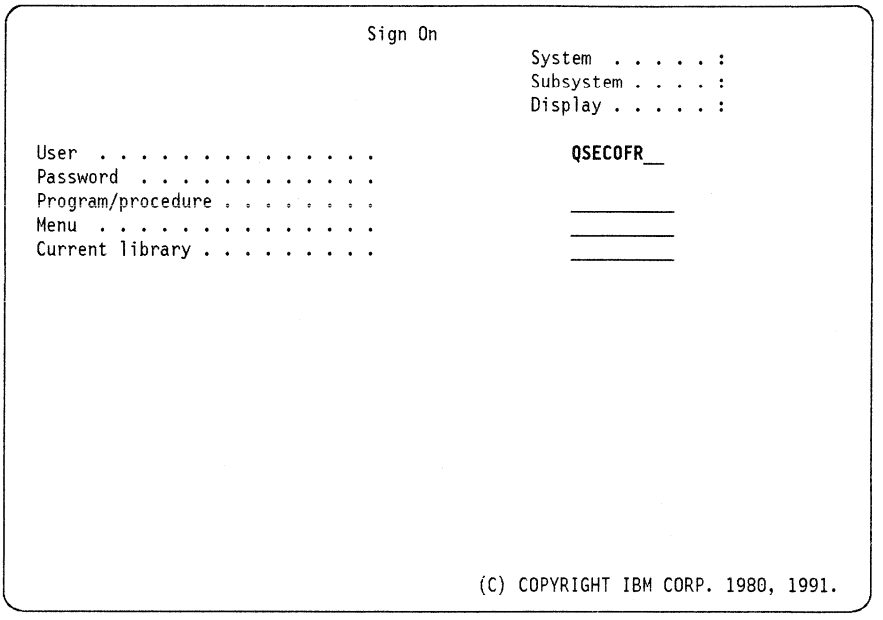


Figure 8-1. Sign On Display

2. To change the QSECOFR password, type CHGPWD on the command line of a menu, and press the Enter key.

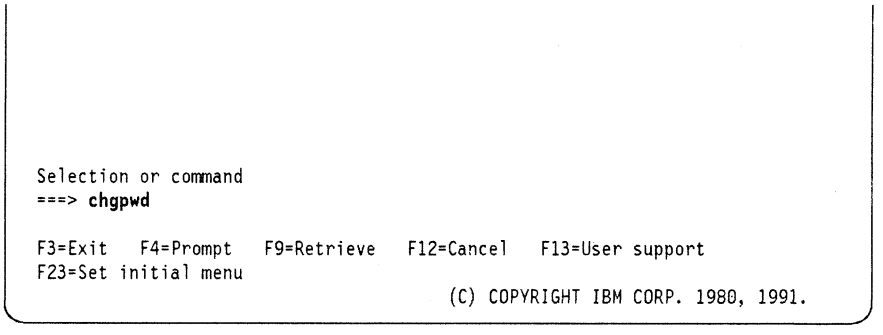


Figure 8-2. Using the Change Password Command

3. Type QSECOFR for the *Current password* prompt, your new password for the *New password* prompt, and your new password again for the *New password (to verify)* prompt. Press the Enter key. This password should only be known to you and the user that you have assigned as your alternative security officer.

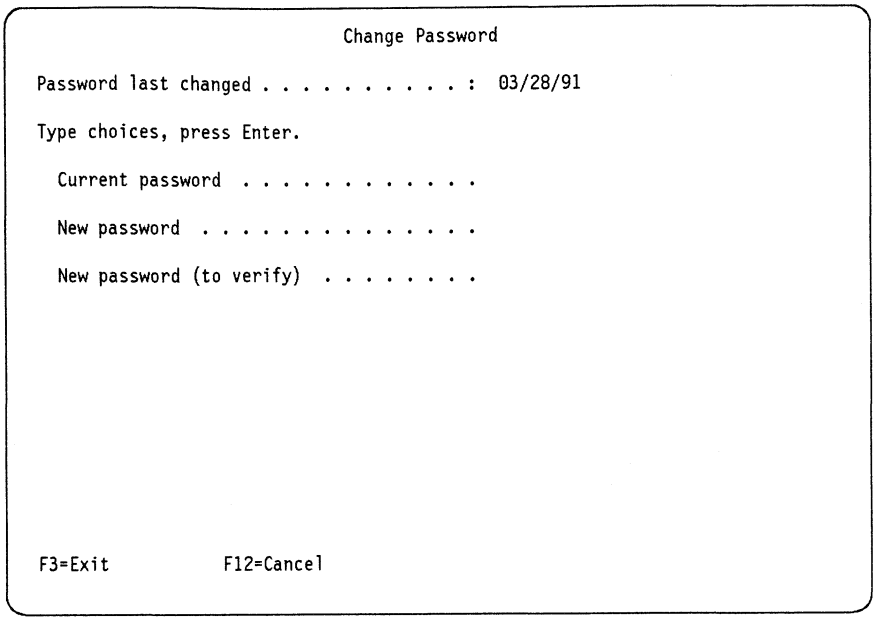
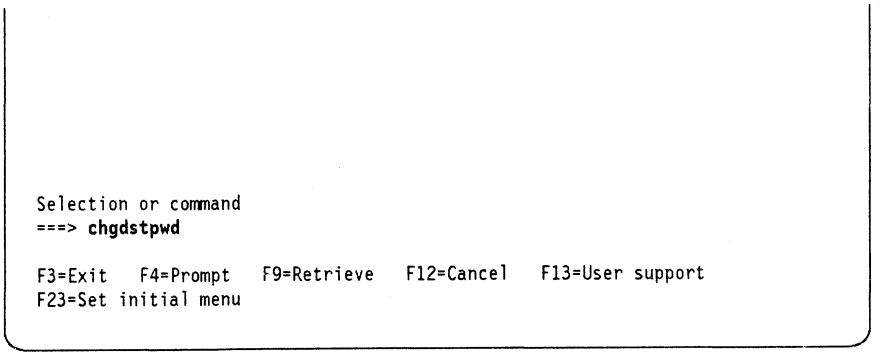


Figure 8-3. Changing the Security Officer's Password

Resetting the Dedicated Service Tools (DST) Passwords to the System-Supplied Default

It may be necessary to reset the DST passwords back to the system-supplied defaults. Resetting the passwords affect full, basic, and security capability passwords.

1. Type the Change DST Password (CHGDSTPWD) command on the command line of a menu and press F4 (Prompt). You must be the security officer (QSECOFR) user profile to use the CHGDSTPWD command.



2. Type *DEFAULT in the *DST security officer password* prompt and press the Enter key.

```
                    Change DST Password (CHGDSTPWD)

Type choices, press Enter.

DST security officer password  .  *DEFAULT     *SAME, *DEFAULT
```

The DST passwords are reset to the system-supplied defaults.

Changing the DST Passwords Using DST

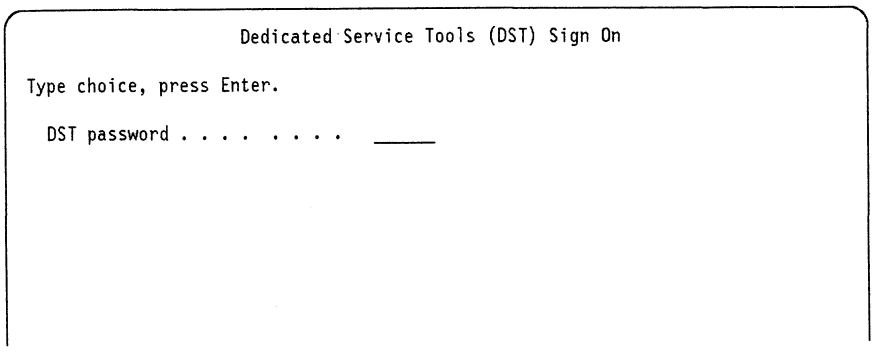
You must provide the DST security password to change DST passwords. You can change the DST basic, full, or security passwords using the DST Password menu. You cannot change these passwords from any OS/400 menu.

To change DST passwords, do the following.

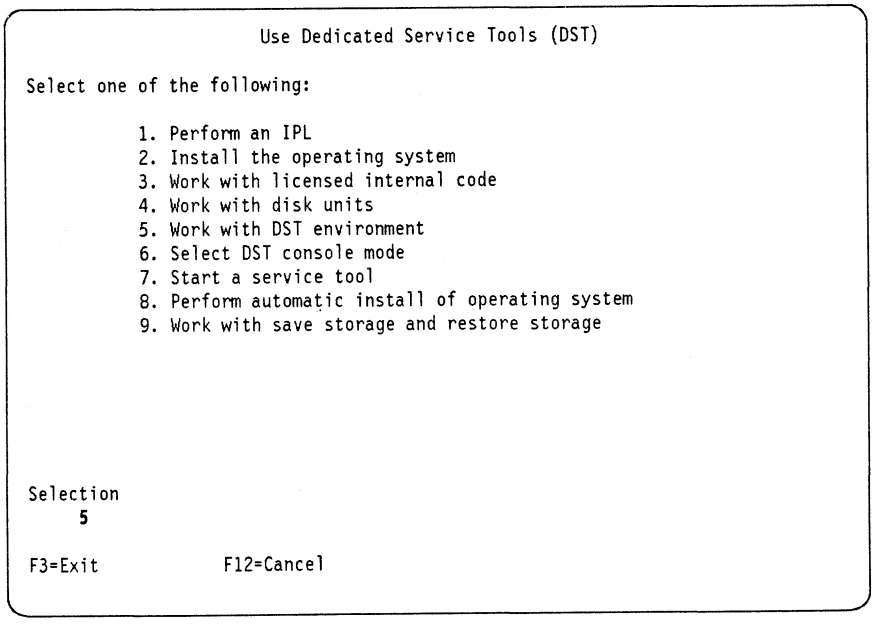
1. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST) and press the Enter key.

```
                        IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools
          4. Perform automatic install of the operating system









Selection
     3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                          © COPYRIGHT IBM CORP. 1980, 1991.
```

2. Sign on DST with the DST *security* level password.



3. Select option 5 (Work with DST environment) on the Use Dedicated Service Tools (DST) display and press the Enter key.



4. Select option 9 (Change DST password) on the Work with DST Environment display and press the Enter key.

```
                        Work with DST Environment

Select one of the following:

     1. Work with active service tools
     2. Select output printer
     3. Cancel printer output
     4. Cancel printer output and deallocate printer
     5. Select tape
     6. Cancel tape operation and deallocate tape
     7. Select diskette
     8. Cancel diskette operation and deallocate diskette
     9. Change DST passwords




Selection
     9

F3=Exit          F12=Cancel
```

5. Select the option for the password you want to change. If you want to reset the operating system security officer (QSECOFR) user profile password to the system-supplied default, select option 4, (Reset system default password).

```
                          Change DST Password

Select one of the following:

          1. Change the DST basic capability password
          2. Change the DST full capability password
          3. Change the DST security capability password
          4. Reset system default password
          5. Change the operating system install security




Selection
     2

F3=Exit          F12=Cancel
```

6. Enter the current password, and then the new password twice to verify. Press the Enter key.

```
                    Change DST Full Capability Password

Type choices, press Enter.

 Full capability

   Current password . . . . . . . . . . . . . . . _______

   New password . . . . . . . . . . . . . . . . . _______

   New password . . . . . . . . . . . . . . . . . _______

F3=Exit            F12=Cancel
```

Changing System Install Security

System install security is the method of defining who can install the operating system on your AS/400 system.

To prevent certain users from installing the system you need to use the system install security task. To use the operating system install security task do the following:

1. From the IPL or Install the System menu, select option 3 (Use Dedicated Service Tools (DST)) and press the Enter key.

```
                         IPL or Install the System

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Use Dedicated Service Tools
          4. Perform automatic install of the operating system




Selection
    3

Licensed Internal Code is subject to the licensed
granted in the Agreement for Purchase of IBM machines.
                                        © COPYRIGHT IBM CORP. 1980, 1991.
```

2. Sign on DST with the DST *security* level password.

```
                    Dedicated Service Tools (DST) Sign On

Type choice, press Enter.

  DST password . . . . . .  _____
```

3. Select option 5 (Work with DST environment) on the Use Dedicated Service Tools (DST) display and press the Enter key.

```
                        Use Dedicated Service Tools (DST)

Select one of the following:

          1. Perform an IPL
          2. Install the operating system
          3. Work with licensed internal code
          4. Work with disk units
          5. Work with DST environment
          6. Select DST console mode
          7. Start a service tool
          8. Perform automatic install of operating system
          9. Work with save storage and restore storage




Selection
     5

F3=Exit           F12=Cancel
```

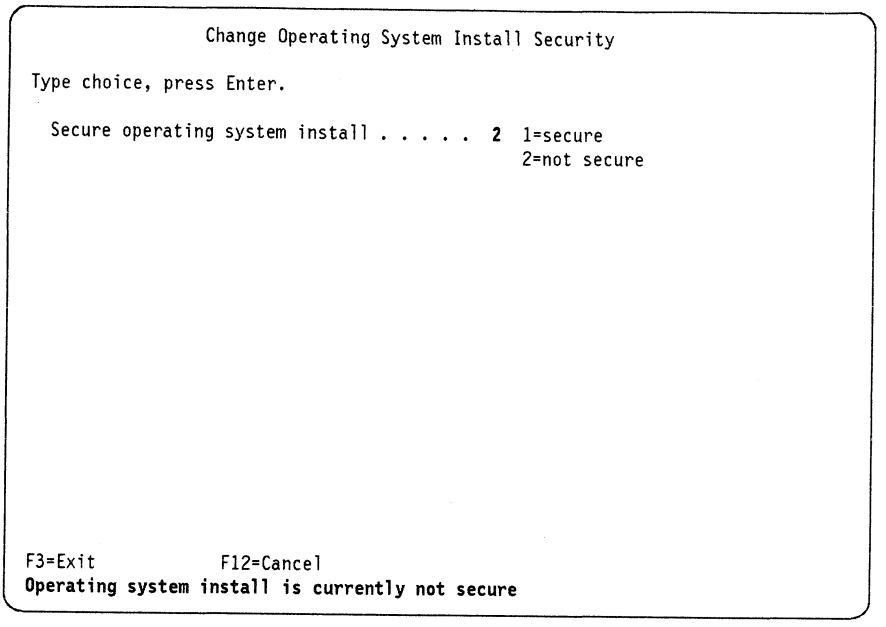
4. Select option 9 (Change DST password) on the Work with DST Environment display and press the Enter key.

```
                        Work with DST Environment

Select one of the following:

     1. Work with active service tools
     2. Select output printer
     3. Cancel printer output
     4. Cancel printer output and deallocate printer
     5. Select tape
     6. Cancel tape operation and deallocate tape
     7. Select diskette
     8. Cancel diskette operation and deallocate disk
     9. Change DST passwords




Selection
     9

F3=Exit           F12=Cancel
```

5. Select option 5 (Change the operating system install security) on the Change DST Password display and press the Enter key.

```
                              Change DST Password

Select one of the following:

          1. Change the DST basic capability password
          2. Change the DST full capability password
          3. Change the DST security capability password
          4. Reset system default password
          5. Change the operating system install security




Selection
     5

F3=Exit           F12=Cancel
```

6. Type a 1 (Secure) or a 2 (Not secure) in the *Change the operating system install security* field. Press the Enter key. A message is shown at the bottom of the display showing the new security level.



7. Press F3 (Exit) to return to the DST main menu. The change to the operating system install security is effective

Working with System Values That Affect Security

You can change the system values that affect security by using an option on a menu or by using the Work with system Value (WRKSYSVAL) command. These values are discussed in the topic "Determine the System Values to Use" on page 7-12.

1. To change the system values, do the following:
   a. Select option 7 (Define or change the system) from the AS/400 Main Menu, and press the Enter key
   b. Select option 8 (Work with System Values) and press the Enter key.

   To bypass the menus, type WRKSYSVAL on the command line, and press F4 (Prompt).

   The Work with System Value display is shown.
2. Type *SEC in the *Type of system value* prompt and press the Enter key to display a list of system values related to security.

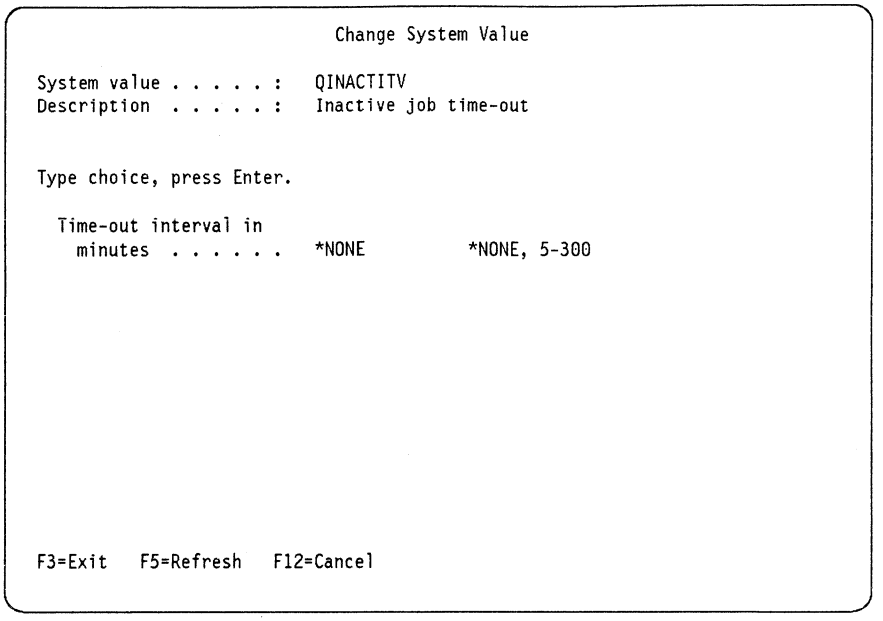
```
                    Work with System Value (WRKSYSVAL)

Type choices, press Enter.

System value . . . . . .  *SEC          *ALL, QABNORMSW, QACGLVL...

Output . . . . . .  *            *, *PRINT

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

3. Use the Page Up and Page Down keys to find the system values you want to change.
4. Type a 2 (Change) in the *Option* column for the system values you want to change, and press the Enter key.

```
                         Work with System Values
                                                          System:   RCH38360
Position to . . . . . .              Starting characters of system value
Subset by Type . . . . .  *SEC       F4 for list

Type options, press Enter.
  2=Change   5=Display

       System
Option Value       Type     Description
       QAUDLVL     *SEC     Security auditing level
       QCRTAUT     *SEC     Create default public authority
       QDSPSGNINF  *SEC     Sign-on display information control
  2    QINACTITV   *SEC     Inactive job time-out
       QINACTMSGQ  *SEC     Inactive job message queue
       QLMTDEVSSN  *SEC     Limit device sessions
       QLMTSECOFR  *SEC     Limit security officer device access
       QMAXSGNACN  *SEC     Action to take for failed signon attempts
                                                                    More...
Command
===>
F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F11=Display names only
F12=Cancel
```

Figure 8-4. Working with System Values Related to Security

5. When the Change System Value display is shown, change the value, and press the Enter key.



Working with User Profiles

If you completed a planning form, User Profile Form (Part 1), for each user of your system, you can use the information on the completed forms when you create the user profiles. If you did not complete a planning form, you can still create user profiles. However, you might want to write some notes about each user before you begin.

The following topics show step-by-step tasks to:

- Create a group profile
- Create a user profile that is a member of a group
- Copy an existing user profile
- Grant a group or user profile authority for an object
- Display or print user profile information
- Delete a user profile that owns objects
- Work with objects by owner
- Display programs that adopt the owner's authority

Creating a Group Profile

You should create your group profiles before the individual user profiles because a user cannot become a member of a group profile unless the group profile already exists. If you have completed your planning, you should know how many group profiles to create and who will be members of each group profile. See Figure 8-5 on page 8-13.

User Profile Form (Part 1)

Name: Group662
Position:
Responsibilities:
Use profile (Yes, No):
Group profile (Yes, No): Yes
Group member (Yes, No):
Group profile name:

The default values are in parentheses.
Required = *

*User: GROUP662	Current library (*CRTDFT)
Password (*USRPRF)	Initial program *NONE)
Set password to expire (*NO)	Initial program (*NONE)
Profile status (*ENABLED)	Library name (*LIBL)
User class (*USER)	Initial menu (MAIN)
Assistance Level (*SYSVAL)	Library name (*LIBL)
	Limited capability (*NO)
	Text (*BLANK): 'Manufacturing group profile'

Additional Parameters:

Special authority (*USRCLS)	Group profile (*NONE)	Output queue (*WRKSTN)
Special environment (*SYSVAL)	Owner (*USRPRF)	Library name (*LIBL)
Display sign-on information (*SYSVAL)	Group authority (*NONE)	Attn-key-handling program (*SYSVAL)
Password expiration interval (*SYSVAL)	Accounting code (*BLANK)	Library name (*LIBL)
Limit device sessions (*SYSVAL)	Document password (*NONE)	Language identifier (*SYSVAL)
Keyboard buffering (*SYSVAL)	Message queue (*USRPRF)	Country identifier (*SYSVAL)
Maximum storage (*NOMAX)	Library name (*LIBL)	Coded character set identifier (*SYSVAL)
Priority limit (3)	Delivery (*NOTIFY)	User options (*NONE)
Job description (QDFTJOBD)	Severity (00)	Authority (*EXCLUDE)
Library name (*LIBL)	Print device (*WRKSTN)	

For Group Profiles ONLY:

Member name	Member name	Member name	Member name
WENDTW	STACKR		
PETERSONJ	CUSHIONM		
BURNS	PACKERDP		

Note: You may copy as necessary.

RV2L054-1

Figure 8-5. Completed User Profile Form

To create a group profile:

1. Type the Create User Profile (CRTUSRPRF) command on the command line of a menu, and press F4 (Prompt).
2. Type the group information (see Figure 8-6 on page 8-14), such as the user profile name, user password, limited capabilities, and text (text must be enclosed in apostrophes).

   The parameters shown in Figure 8-4 on page 8-11 are related to security. If you do not want to specify additional parameters, press the Enter key. If you do want to specify additional parameters (see Figure 8-6 on page 8-14), press F10 (Additional parameters).

```
                    Create User Profile (CRTUSRPRF)

Type choices, press Enter.

User profile . . . . . > group662          Name
User password . . . . . *none           Name, *USRPRF, *NONE
Set password to expired . . . . *NO          *NO, *YES
Status . . . . . *ENABLED      *ENABLED, *DISABLED
User class . . . . . *USER         *USER, *SYSOPR, *PGMR...
Assistance level . . . . . *SYSVAL       *SYSVAL, *BASIC, *INTERMED...
Current library . . . . . *CRTDFT       Name, *CRTDFT
Initial program to call . . . . *NONE         Name, *NONE
  Library . . . . . .                 Name, *LIBL, *CURLIB
Initial menu . . . . . MAIN         Name, *SIGNOFF
  Library . . . . . .   *LIBL       Name, *LIBL, *CURLIB
Limit capabilities . . . . . *USRCLS       *USRCLS, *NO, *PARTIAL, *YES
Text 'description' . . . . . 'Manufacturing group profile'



                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 8-6. Create User Profile Display (Group)

Press F10 to display additional parameters. These parameters are used to tailor a user's environment on the system.

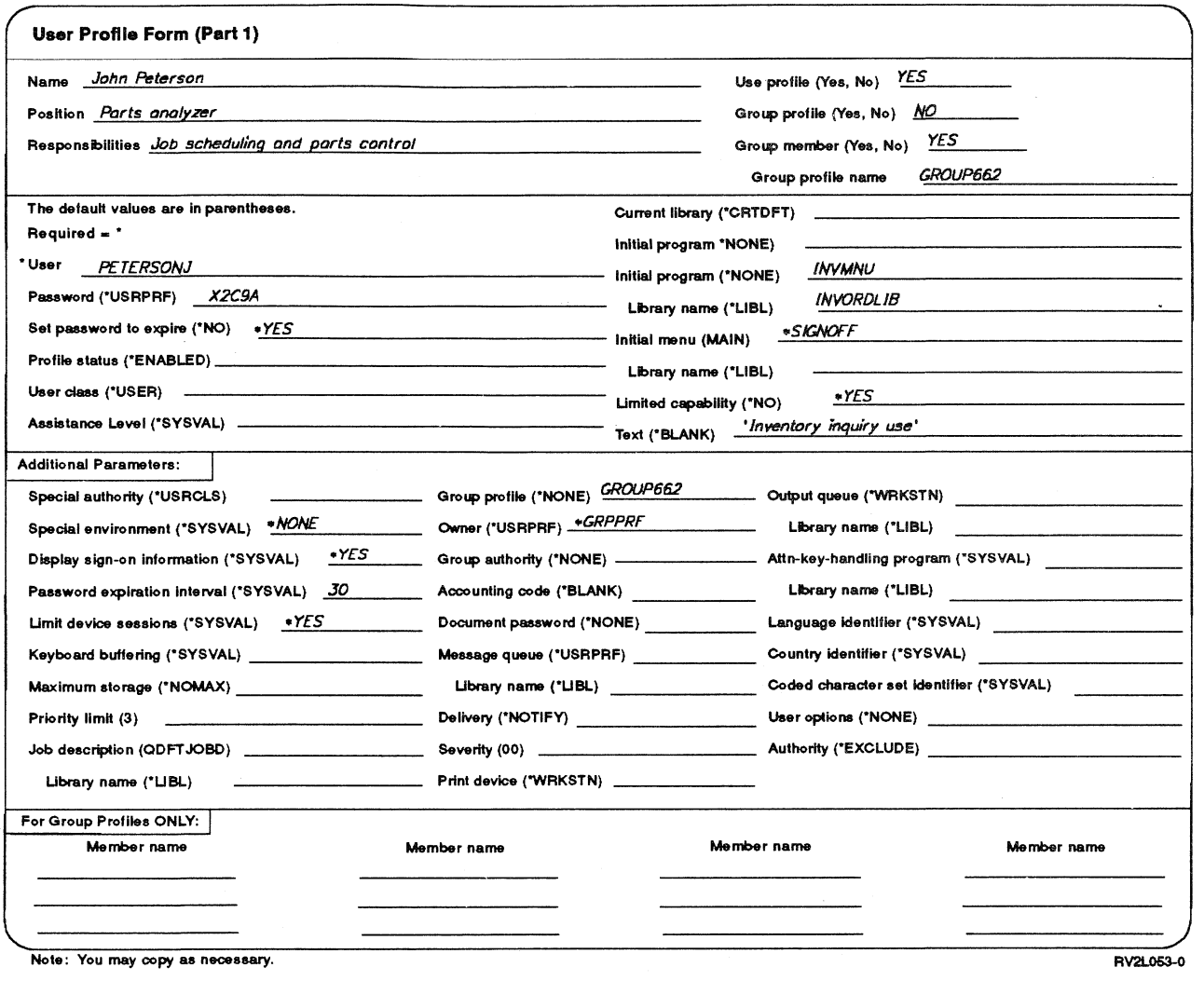
3. Use the Page Down key if you want to see more of the additional parameters. Type in the additional information, and press the Enter key.

You can use the Create User Profile (CRTUSRPRF) command and parameters as an alternative to the prompt displays. To bypass the prompt displays, you can type the command and parameters on the command line of a menu by specifying the group profile name and pressing the Enter key. For example:

```
CRTUSRPRF USRPRF(GROUP662) PASSWORD(*NONE)
      TEXT('Manufacturing group profile')
```

Creating an Individual User Profile

After you have created the group profiles, you can create the individual user profiles. If the user is a member of a group, the group information is specified in the individual user profile, not the group profile. If you have completed your planning, you should know how many user profiles to create and if the user will be a member of a group. See Figure 8-7 on page 8-15.



User Profile Form (Part 1)

Name: John Peterson
Position: Parts analyzer
Responsibilities: Job scheduling and parts control
Use profile (Yes, No): YES
Group profile (Yes, No): NO
Group member (Yes, No): YES
Group profile name: GROUP662

The default values are in parentheses.
Required = *

*User: PETERSONJ
Password (*USRPRF): X2C9A
Set password to expire (*NO): *YES
Profile status (*ENABLED):
User class (*USER):
Assistance Level (*SYSVAL):

Current library (*CRTDFT):
Initial program *NONE):
Initial program (*NONE): INVMNU
Library name (*LIBL): INVORDLIB
Initial menu (MAIN): *SIGNOFF
Library name (*LIBL):
Limited capability (*NO): *YES
Text (*BLANK): 'Inventory inquiry use'

Additional Parameters:

Special authority (*USRCLS):
Special environment (*SYSVAL): *NONE
Display sign-on information (*SYSVAL): *YES
Password expiration interval (*SYSVAL): 30
Limit device sessions (*SYSVAL): *YES
Keyboard buffering (*SYSVAL):
Maximum storage (*NOMAX):
Priority limit (3):
Job description (QDFTJOBD):
Library name (*LIBL):

Group profile (*NONE): GROUP662
Owner (*USRPRF): *GRPPRF
Group authority (*NONE):
Accounting code (*BLANK):
Document password (*NONE):
Message queue (*USRPRF):
Library name (*LIBL):
Delivery (*NOTIFY):
Severity (00):
Print device (*WRKSTN):

Output queue (*WRKSTN):
Library name (*LIBL):
Attn-key-handling program (*SYSVAL):
Library name (*LIBL):
Language identifier (*SYSVAL):
Country identifier (*SYSVAL):
Coded character set identifier (*SYSVAL):
User options (*NONE):
Authority (*EXCLUDE):

For Group Profiles ONLY:

Member name | Member name | Member name | Member name

Note: You may copy as necessary.

RV2L053-0

Figure 8-7. Completed User Profile Form (Part 1)

To create an individual user profile:

1. Type CRTUSRPRF on the command line of a menu and press F4 (Prompt).
2. Type the user profile information (see Figure 8-8), such as the user profile name, user password, limited capabilities, and text (text must be enclosed in apostrophes).

```
                    Create User Profile (CRTUSRPRF)

Type choices, press Enter.

User profile . . . . . .  petersonj     Name
User password  . . . . . .  x2c9a        Name, *USRPRF, *NONE
Set password to expired  . . . .  *yes   *NO, *YES
Status . . . . . .  . . . .  *ENABLED    *ENABLED, *DISABLED
User class . . . . . .  . . .  *USER     *USER, *SYSOPR, *PGMR...
Assistance level . . . . . .  *SYSVAL    *SYSVAL, *BASIC, *INTERMED...
Current library  . . . . . .  *CRTDFT    Name, *CRTDFT
Initial program to call  . . . .  invmnu  Name, *NONE
  Library  . . . . . .  . . .    invordlib  Name, *LIBL, *CURLIB
Initial menu . . . . . .  . . .  *signoff  Name, *SIGNOFF
  Library  . . . . . .  . . .              Name, *LIBL, *CURLIB
Limit capabilities . . . . . .  *yes      *USRCLS, *NO, *PARTIAL, *YES
Text 'description' . . . . . .  Inventory inquiry use

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display        F24=More keys
```

Figure 8-8. Creating User Profile Display (Individual)

The parameters shown in Figure 8-8 are related to security. If you do not want to specify additional parameters, press the Enter key.

Press F10 to display additional parameters. These parameters are used to tailor a user's environment on the system.

3. Use the Page Down key if you want to see more of the additional parameters (see Figure 8-9 and Figure 8-10 on page 8-17). If this user profile is a member of a group, specify the group profile information for the *Group profile* and *Group authority* prompts. If you specify *GRPPRF for the *Owner* prompt, you must specify *NONE for the *Group authority* prompt. Type the additional information, and press the Enter key.

```
                    Create User Profile (CRTUSRPRF)

Type choices, press Enter.


                         Additional Parameters

Special authority . . . . . .  *USRCLS      *USRCLS, *NONE, *SAVSYS...
            + for more values
Special environment . . . . . .  *none      *SYSVAL, *NONE, *S36
Display sign-on information  . .  *yes      *SYSVAL, *NO, *YES
Password expiration interval . .  30        1-366, *SYSVAL, *NOMAX
Limit device sessions  . . . . .  *yes      *SYSVAL, *YES, *NO
Keyboard buffering . . . . . .  *SYSVAL     *SYSVAL, *NO, *TYPEAHEAD...
Maximum allowed storage  . . . .  *NOMAX    Kilobytes, *NOMAX
Highest schedule priority  . . .  3         0-9
Job description  . . . . . .  QDFTJOBD      Name
  Library  . . . . . .  . . .    *LIBL       Name, *LIBL, *CURLIB
Group profile  . . . . . .  group662        Name, *NONE
Owner  . . . . . .  . . . . .  *grpprf      *USRPRF, *GRPPRF
                                                                    More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 8-9. Create User Profile Display (Additional Parameters)

```
                    Create User Profile (CRTUSRPRF)

Type choices, press Enter.

Group authority . . . . . .  *NONE         *NONE, *ALL, *CHANGE, *USE...
Accounting code . . . . . .  *BLANK
Document password . . . . . .  *NONE       Name, *NONE
Message queue . . . . . .  *USRPRF         Name, *USRPRF
  Library . . . . . .                      Name, *LIBL, *CURLIB
Delivery . . . . . .  *NOTIFY              *NOTIFY, *BREAK, *HOLD, *DFT
Severity code filter . . . . . .  0        0-99
Print device . . . . . .  *WRKSTN          Name, *WRKSTN, *SYSVAL
Output queue . . . . . .  *WRKSTN          Name, *WRKSTN, *DEV
  Library . . . . . .                      Name, *LIBL, *CURLIB
Attention program . . . . . .  *SYSVAL    Name, *NONE, *SYSVAL, *ASSIST
  Library . . . . . .                      Name, *LIBL, *CURLIB
Language ID . . . . . .  *SYSVAL           Character value, *SYSVAL
Country ID . . . . . .  *SYSVAL            Character value, *SYSVAL
Coded Character Set ID . . . . .  *SYSVAL  *SYSVAL, *HEX, F4 for list

                                                                   More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

```
                    Create User Profile (CRTUSRPRF)

Type choices, press Enter.

User options . . . . . .  *NONE            *NONE, *CLKWD, *EXPERT...
             + for more values
Authority . . . . . .  *EXCLUDE           *ALL, *CHANGE, *USE, *EXCLUDE

                                                                   Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Figure 8-10. (For Version 2 Release 1.1) Create User Profile Display (Additional Parameters)

You can use the Create User Profile (CRTUSRPRF) command and parameters as an alternative to the prompt displays. To bypass the prompt displays, you can type the command and parameters on the command line of a menu by specifying the user profile name and the group profile information and pressing the Enter key. For example:

```
CRTUSRPRF USRPRF(PETERSONJ) PASSWORD(X2C9A) PWDEXP(*YES)
     INLPGM(INVORDLIB/INVMNU) INLMNU(*SIGNOFF)
     LMTCPB(*YES) TEXT('Inventory inquiry use') SPCENV(*NONE)
     DSPSGNINF(*YES) PWDEXPITV(30) LMTDEVSSN(*YES) GRPPRF(GROUP662)
     OWNER (*GRPPRF)
```

Copying an Existing User Profile

When creating user profiles, you may have a user profile that already exists that can be used as a base for other user profiles. You can copy the base profile by using the Work with User Profiles (WRKUSRPRF) command. WRKUSRPRF is an interactive function only. See CPYUSRPRF command in library QUSRTOOL for an example of how to copy a user profile using a command interface.

If you copy an existing user profile using WRKUSRPRF, only the user profile attributes are copied. The base user profile has specific authorities to objects that are not copied to the new user profile. The new user profile is not automatically enrolled in any licensed programs.

To copy an existing user profile:

1. Type WRKUSRPRF on the command line of a menu and press F4 (Prompt).
2. When the following display appears, type the name of the user profile you want to copy in the *User profile* prompt or type *ALL to display all user profiles that you have authority to.

```
                    Work with User Profiles (WRKUSRPRF)

Type choices, press Enter.

User profile . . . . . .  petersonj       Name, generic*, *ALL
```

3. Press the Enter key. The following display is shown:

```
                          Work with User Enrollment
                                                              System:   RCH38360
Type options below, then press Enter.
  1=Add   2=Change   3=Copy   4=Remove   5=Display

Opt     User          Description

 3      PETERSONJ     Inventory inquiry use

                                                                          Bottom
F1=Help      F3=Exit   F5=Refresh   F9=Command line   F12=Cancel   F17=Top
F18=Bottom   F21=Select assistance level
```

4. Type a 3 (Copy) in the *Option* field and press the Enter key.

5. The Copy User display is shown. Type the name of the new user in the *User profile* prompt. If you want to change any of the other values, do not press the Enter key.

```
                                Copy User

Copy from user . . . . :   PETERSONJ

Type choices below, then press Enter.

User . . . . . .  .  .  .  THOMASP          Name
User description . . . .   Pat Thomas Dept 547 6165
Password . . . . . .  .  .  T2M9F           Characters
Type of user . . . . . .   *USER            Type, F4 for list
User group . . . . . .  .  GROUP662         Name, F4 for list

Restrict command line use  Y               Y=Yes, N=No
Uses OfficeVision/400 . .  N               Y=Yes, N=No

Default library . . . . .                  Name
Default printer . . . . .  *WRKSTN          Name, *WRKSTN, F4 for list
First menu . . . . . .  .  *SIGNOFF        Name
  Library . . . . . .  .  .                Name


                                                                    More...
 F1=Help   F3=Exit   F5=Refresh   F12=Cancel
```

6. If you want to change any of the other values, use the Page Up and Page Down keys and enter the new values.

```
                                Copy User

Copy from user . . . . :   PETERSONJ

Type choices below, then press Enter.

Sign on program . . . . .  INVMNU          Name, *NONE
  Library . . . . . .  .  .    INVORDLIB     Name

Attention key program . .  *SYSVAL         Name, *SYSVAL, *ASSIST, *NONE
  Library . . . . . .  .  .                Name

Option 50 on OfficeVision/400 menu:
  Text for menu option
  User program . . . . .                   Name
    Library . . . . . .  .                 Name




                                                                    Bottom
 F1=Help   F3=Exit   F5=Refresh   F12=Cancel
```

7. When you have finished changing the values, press the Enter key. The following display is shown.

```
                          Work with User Enrollment
                                                              System:   RCH38360
Type options below, then press Enter.
  1=Add   2=Change   3=Copy   4=Remove   5=Display

Opt     User           Description

        PETERSONJ      Inventory inquiry use

                                                                          Bottom
F1=Help      F3=Exit   F5=Refresh   F9=Command line   F12=Cancel   F17=Top
F18=Bottom   F21=Select assistance level
User THOMASP copied from user PETERSONJ.
```

The new user profile THOMASP was created with no special authorities.

Granting Group and User Profile Authority to Objects

If you completed User Profile Form (Part 2), Resource Security for each group profile and individual user profile, you can use the information on the completed forms when you grant a group profile or a user profile authority for objects.

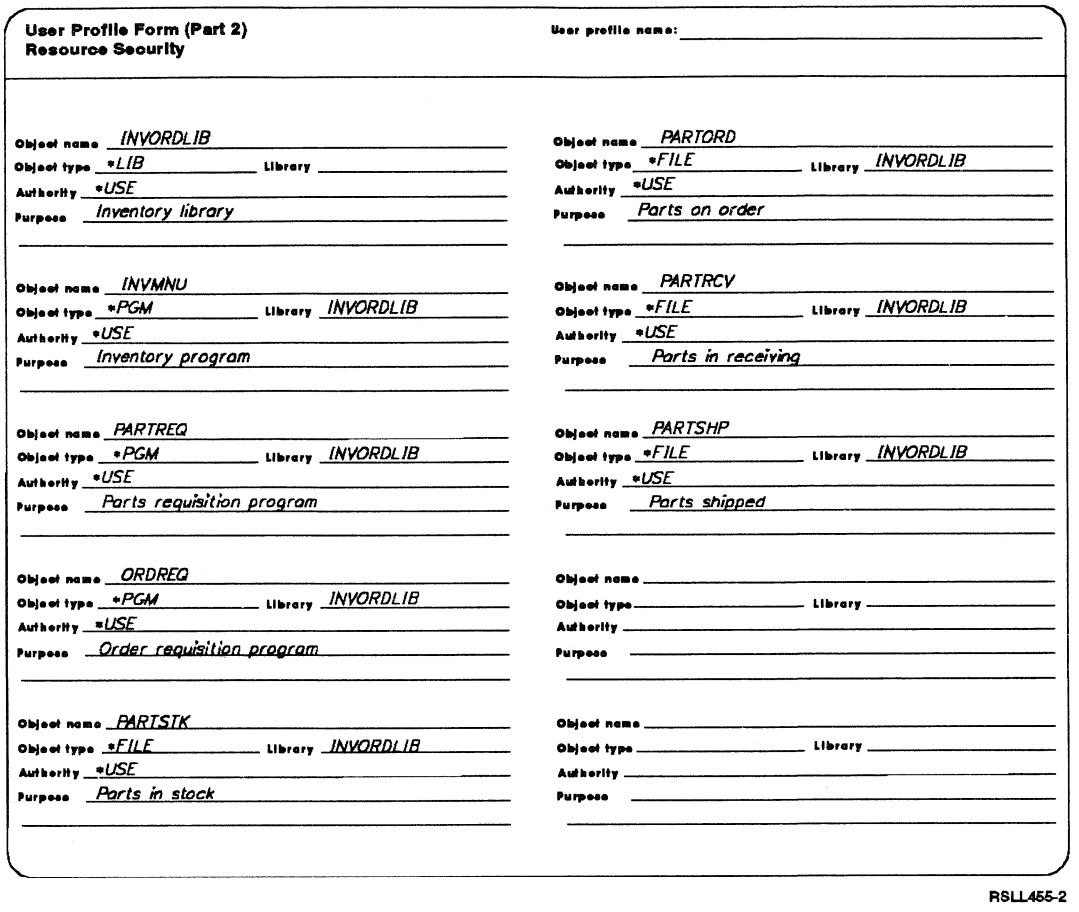


Figure 8-11. User Profile Form (Part 2)

You can use the Edit Object Authority (EDTOBJAUT) command (interactive only) or the Grant Object Authority (GRTOBJAUT) command (interactive or batch) to grant group or user profiles authority for objects. In this example, the EDTOBJAUT command is used.

1. Type EDTOBJAUT on the command line of a menu and press F4 (Prompt).
2. Type the name of the object for the *Object* prompt, the name of the library where the object is located for the *Library* prompt, and the type of object for the *Object type* prompt. Press the Enter key.

```
                    Edit Object Authority (EDTOBJAUT)

Type choices, press Enter.

Object . . . . . .  invmnu          Name
  Library . . . . . .   invordlib    Name, *LIBL, *CURLIB
Object type . . . . . .  *pgm       *AUTL, *CFGL, *CHTFMT...
```

Figure 8-12. Edit Object Authority Display

The following display shows the name of the object owner and the users who have authority to the object. If the object is secured by an authorization list, the authorization list name is shown.

```
                          Edit Object Authority

Object . . . . . :   INVMNU          Object type  . . . . :   *PGM
  Library  . . . . :     INVORDLIB     Owner  . . . . . . . :   SMITHP

Type changes to current authorities, press Enter.

  Object secured by authorization list  . . . . . . . . . . . .   *NONE

             Object    ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
SMITHP       *ALL        X    X     X     X     X     X       X
*PUBLIC      *CHANGE     X                X     X     X       X

                                                                    Bottom
F3=Exit   F5=Refresh  F6=Add new users  F10=Grant with reference object
F11=Nondisplay detail F12=Cancel        F17=Top   F18=Bottom
```

Figure 8-13. Edit Object Authority Display

3. To grant a user or group authority to the object, press F6 (Add new users). Type the user or group profile name in the *User* column and the authority you are giving the user or group in the *Object Authority* column or in the *Object* and *Data* columns. Press the Enter key.

```
                             Add New Users

Object . . . . . :   INVMNU          Object type  . . . . :   *PGM
  Library  . . . . :     INVORDLIB     Owner  . . . . . . . :   SMITHP

Type new users, press Enter.

             Object    ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
petersonj    *use

                                                                    More...
F3=Exit   F11=Nondisplay detail   F12=Cancel  F17=Top  F18=Bottom
```

Figure 8-14. Edit Object Authority Display

After you press the Enter key, the user or group profile name that you gave authority to is shown on the display. See Figure 8-15 on page 8-23.

```
                          Edit Object Authority

Object . . . . . :   INVMNU          Object type  . . . . :   *PGM
  Library  . . . . :     INVORDLIB     Owner  . . . . . . . :   SMITHP

Type changes to current authorities, press Enter.

  Object secured by authorization list  . . . . . . . . . . . .   *NONE

             Object    ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
PETERSONJ    *USE        X                X
SMITHP       *ALL        X    X     X     X     X     X       X
*PUBLIC      *CHANGE     X                X     X     X       X









                                                                  Bottom
F3=Exit   F5=Refresh  F6=Add new users  F10=Grant with reference object
F11=Nondisplay detail F12=Cancel        F17=Top   F18=Bottom
```

Figure 8-15. Edit Object Authority Display

You can use the Grant Object Authority (GRTOBJAUT) command and parameters as an alternative to the prompt displays. To bypass the prompt displays, you can type the command and parameters on the command line of a menu, and press the Enter key. For example:

```
GRTOBJAUT OBJ(INVORDLIB/INVMNU) OBJTYPE(*PGM)
      USER(PETERSONJ) AUT(*USE)
```

To grant the group authority for all types of objects in library INVORDLIB, enter:

```
GRTOBJAUT OBJ(INVORDLIB/*ALL) OBJTYPE(*ALL)
      USER(GROUP662) AUT(*USE)
```

Displaying and Printing User Profile Information

The Display User Profile (DSPUSRPRF) command displays the contents of a user profile. The user profile contains the user's operational limits for system resources: the names of commands, devices, and objects the user owns or has authority to use.

The Display User Profile command can also be used to print the user profile or copy some of the user profile information into a database file. The types of information copied into the database file are basic information (*BASIC), objects owned by the user (*OBJOWN), and objects the user is authorized to use (*OBJAUT). The information copied into a database file can be used to manage security and perform auditing tasks.

1. To display a user profile, type DSPUSRPRF on the command line of a menu and press the F4 (Prompt) key.
2. Type the user profile name for the *User profile* prompt and the type of information for the *Type of information* prompt. If *GENERIC or *ALL is specified for the user profile name, the TYPE(*BASIC) and OUTPUT(*OUTFILE) must also be specified.

3. Type one of the following for the *Output* prompt:

   * To see the information at the display station

   ***PRINT** To print the information

   ***OUTFILE** To copy the information to a database file (*OUTFILE is valid for information types *BASIC, *OBJOWN or *OBJAUT)

   Press the Enter key.

```
                        Display User Profile (DSPUSRPRF)

Type choices, press Enter.

User profile . . . . . .  smithp       Name, *GENERIC, *ALL
Type of information  . . .  *BASIC     *BASIC, *ALL, *CMDAUT...
Output . . . . . . . . . .  *          *, *PRINT, *OUTFILE
```

*Figure 8-16. Display User Profile (*BASIC)*

Information for the following displays is shown if *BASIC is specified. The basic information shows the values specified for the user profile. It does not show owned or authorized objects.

```
                        Display User Profile - Basic

User profile . . . . . . . . . . . . . . :  SMITHP

Previous sign-on . . . . . . . . . . . . :
Sign-on attempts not valid . . . . . . . :  0
Status . . . . . . . . . . . . . . . . . :  *ENABLED
Date password last changed . . . . . . . :  11/29/90
Password expiration interval . . . . . . :  60
  Date password expires  . . . . . . . . :    01/28/91
Set password to expired  . . . . . . . . :  *NO
User class . . . . . . . . . . . . . . . :  *SECOFR
Special authority  . . . . . . . . . . . :  *ALLOBJ
                                            *JOBCTL
                                            *SAVSYS
                                            *SECADM
                                            *SERVICE
                                            *SPLCTL
Group profile  . . . . . . . . . . . . . :  *NONE
                                                                  More...
Press Enter to continue.

F3=Exit   F12=Cancel
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Figure 8-17. Display User Profile with Basic Information

```
                    Display User Profile - Basic

User profile . . . . . :   SMITHP

Owner . . . . . :   *USRPRF
Group authority . . . . . :   *NONE
Assistance level . . . . . :   *SYSVAL
Current library . . . . . :   *CRTDFT
Initial menu . . . . . :   MAIN
  Library . . . . . :     *LIBL
Initial program . . . . . :   QCMD
  Library . . . . . :     *LIBL
Limit capabilities . . . . . :   *NO
Text . . . . . :

Display sign-on information . . . . . :   *SYSVAL
Limit device sessions . . . . . :   *SYSVAL
Keyboard buffering . . . . . :   *SYSVAL

                                                                  More...
Press Enter to continue.

F3=Exit   F12=Cancel
```

Figure 8-18. Display User Profile with Basic Information

```
                    Display User Profile - Basic

User profile . . . . . :   SMITHP

Maximum storage allowed . . . . . :   *NOMAX
  Storage used . . . . . :     8
Highest scheduling priority . . . . . :   3
Job description . . . . . :   QDFTJOBD
  Library . . . . . :     QGPL
Accounting code . . . . . :
Message queue . . . . . :   SMITHP
  Library . . . . . :     QUSRSYS
Message queue delivery . . . . . :   *BREAK
Message queue severity . . . . . :   00
Output queue . . . . . :   *WRKSTN
  Library . . . . . :
Printer device . . . . . :   *WRKSTN
Special environment . . . . . :   *SYSVAL

                                                                  More...
Press Enter to continue.

F3=Exit   F12=Cancel
```

Figure 8-19. Display User Profile with Basic Information

```
                    Display User Profile - Basic

User profile . . . . . :   SMITHP

Attention program . . . . . :   QCL
  Library . . . . . :     *LIBL
Language identifier . . . . . :   *SYSVAL
Country identifier . . . . . :   *SYSVAL
Coded character set identifier . . . . . :   *SYSVAL
User options . . . . . :   *EXPERT

                                                                  Bottom
Press Enter to continue.

F3=Exit    F12=Cancel
```

Figure 8-20. (For Version 2 Release 1.1) Display User Profile with Basic Information

To bypass the prompt display, you can enter the following command from the command line of a menu:

```
DSPUSRPRF USRPRF(SMITHP)
```

4. Information for the following displays is shown in addition to the basic information if TYPE(*ALL) is specified. Use the Page Down key if you want to see all the additional information.

```
                    Display Authorized Commands

User profile . . . . . :   SMITHP

  (User does not have specific authority to any commands.)
```

Figure 8-21. Display User Profile with All Information

```
                    Display Authorized Devices

User profile . . . . . :   SMITHP

  (User does not have specific authority to any devices.)
```

Figure 8-22. Display User Profile with All Information

```
                        Display Authorized Objects

User profile . . . . . :   SMITHP

                                    ----Object-----  -------Data--------
Object      Library     Type        Opr  Mgt  Exist  Read  Add  Upd  Dlt
SMITHP      QSYS        USRPRF       X    X          X     X    X    X
SNOWHITE    QUSRSYS     MSGQ         X    X     X    X     X    X    X
```

Figure 8-23. Display User Profile with All Information

```
                          Display Owned Objects

User profile . . . . . . . :   SMITHP
Total objects  . . . . . . :   12

                                                 Authority
Object          Library         Type              Holder
INVMNU          INVORDLIB       FILE
PARTORD         INVORDLIB       FILE
PARTRCV         INVORDLIB       FILE
PARTSHP         INVORDLIB       FILE
PARTSTK         INVORDLIB       FILE
INVLST          QSYS            AUTL
INVORDLIB       QSYS            LIB
SMITHP          QSYS            LIB
NEWSCREE        SMITHP          FILE
SCREENS         SMITHP          FILE
SMITHP          SMITHP          FILE
U1              SMITHP          FILE
                                                                    Bottom
Press Enter to continue.

F3=Exit   F12=Cancel  F17=Top  F18=Bottom
```

Figure 8-24. Display User Profile with All Information

You can bypass the prompt display by entering the following command from the command line of a menu to print all the user profile information:

```
DSPUSRPRF USRPRF(SMITHP) TYPE(*ALL) OUTPUT(*PRINT)
```

5. To copy the owned objects information into a database file, type *OBJOWN for the *Type of information* prompt and *OUTFILE for the *Output* prompt. Press the Enter key.

```
                    Display User Profile (DSPUSRPRF)

Type choices, press Enter.

User profile . . . . . .   smithp          Name, *GENERIC, *ALL
Type of information  . . .   *objown       *BASIC, *ALL, *CMDAUT...
Output . . . . . . . . . .   *outfile      *, *PRINT, *OUTFILE
```

Figure 8-25. Display User Profile Using Outfile

6. Type the name of the file that is to receive the output for the *File to receive output* prompt and the name of the library where the file will be placed for

the *Library* prompt. If you do not want to specify output member options, press the Enter key.

If you do want to specify member options, type a member name for the *Member to receive output* prompt and specify for the *Replace or add records* prompt whether you want to add the information to the member or to replace the information in the member. If you are using a security audit file, you can specify that the records be added to the member.

```
                        Display User Profile (DSPUSRPRF)

Type choices, press Enter.

User profile . . . . . > SMITHP        Name, *GENERIC, *ALL
Type of information . . . . . .  *OBJOWN      *BASIC, *ALL, *CMDAUT...
Output . . . . . > *OUTFILE     *, *PRINT, *OUTFILE
File to receive output . . . . .  audit1       Name, *NONE
  Library . . . . . .  security    Name, *LIBL, *CURLIB
Output member options:
  Member to receive output . . .  *FIRST        Name, *FIRST
  Replace or add records . . . .  *add          *REPLACE, *ADD

                                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh  F12=Cancel   F13=How to use this display
F17=Top   F18=Bottom
```

*Figure 8-26. Display User Profile (*OUTFILE)*

You can bypass the prompt display by entering the following command on the command line of a menu:

```
DSPUSRPRF USRPRF(SMITHP) TYPE(*OBJOWN) OUTPUT(*OUTFILE)
   OUTFILE(SECURITY/AUDIT1) OUTMBR(*FIRST *ADD)
```

The OUTFILE parameter is valid only with a TYPE of *BASIC, *OBJAUT or *OBJOWN.

Deleting a User Profile That Owns Objects

The Delete User Profile (DLTUSRPRF) command allows a user profile to be deleted even if it owns objects. Owned objects can be transferred to another user profile or the objects can be deleted.

Consider the following when deleting a user profile and the owned object value is specified as *DLT:

- If a user profile owns an object type of *USRPRF, the object type *USRPRF will not be deleted from the system. Instead, the ownership of *USRPRF is transferred to the default owner (QDFTOWN) user profile.
- An object type of *LIB is not deleted if the library contains objects owned by other user profiles. Instead, ownership of the library is transferred to the default owner (QDFTOWN) user profile.
- An object type of *RCT is not deleted. Instead, ownership of the object is transferred to the default owner (QDFTOWN) user profile.

- An object type of *PRDDFN is not deleted. Instead, ownership of the object is transferred to the QSYS user profile.
- All rules that apply to the Delete Library (DLTLIB) or other delete commands also apply.

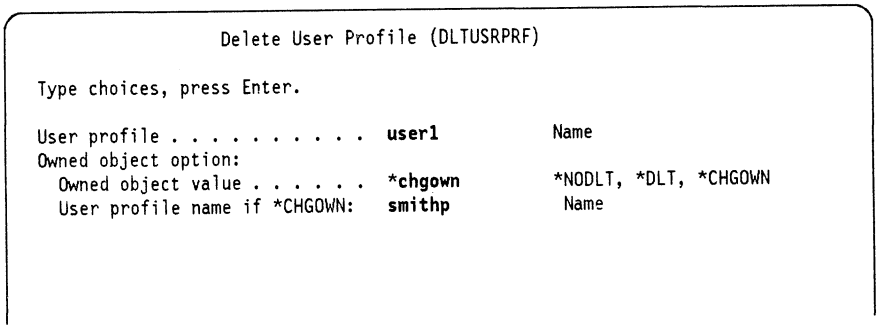
The system determines if the user is enrolled in AS/400 Office. If the user is enrolled in Office, none of the Office objects change ownership or are deleted. A user must be removed from Office before the user profile can be deleted. A message is sent indicating that the user profile cannot be deleted because the user is enrolled in Office.

If an external object is not in a library, then ownership is transferred to the default owner (QDFTOWN) user profile.

1. To delete a user profile, type DLTUSRPRF on the command line of a menu, and press the F4 (Prompt) key.
2. Type the name of the user profile in the *User profile* prompt and the option you want for the *Owned object value* prompt. Press the Enter key.

   If you specify *CHGOWN for the owned objects value, you must specify a user profile name. Ownership of all the objects (except device descriptions) is transferred to the user profile specified.

   If you specify *DLT for the *Owned object value* prompt, then all the objects owned by the user profile are deleted except for the special cases previously listed. The user profile is deleted if all objects process successfully.



Working with Objects by Owner

The Work with Objects by Owner (WRKOBJOWN) command allows a user to change or delete objects owned by any user profile.

1. To work with objects by owner, type WRKOBJOWN on the command line of a menu, and press the F4 (Prompt) key.
2. Type the name of the user profile for the *User profile* prompt, and press the Enter key.

```
                Work with Objects by Owner (WRKOBJOWN)

Type choices, press Enter.

User profile . . . . . .   smithp          Name, *CURRENT
```

3. Type the options you want in the *Opt* column, and press the Enter key.

```
                          Work with Objects by Owner

User profile . . . . . :   SMITHP

Type options, press Enter.
  2=Edit authority        4=Delete   5=Display authority   7=Rename
  8=Display description   9=Change owner

Opt  Object      Library     Type      Attribute   Text
4    INVMNU      INVORDLIB   *FILE     PF          Inventory menu
     PARTORD     INVORDLIB   *FILE     PF          Parts on order
4    PARTRCV     INVORDLIB   *FILE     PF          Parts receivable
4    PARTSHP     INVORDLIB   *FILE     PF          Parts shipped
     INVORDLIB   QSYS        *LIB      PROD        Inventory library
     SMITHP      QSYS        *LIB      TEST        Security test library


                                                                      More...
Parameters or command
===>
F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve  F12=Cancel  F17=Top
F18=Bottom
```

4. The Confirm Delete of Objects display shows a list of objects that you selected for the delete operation. The display allows the user to confirm the choices or return to the previous display to change the objects selected.

```
                       Confirm Delete of Objects

 User profile . . . . . :   SMITHP

Press Enter to confirm your choices for 4=Delete.
Press F12 to return to change your choices.

Option  Object      Library     Type      Attribute   Text
 4      INVMNU      INVORDLIB   *FILE     PF          Inventory menu
 4      PARTRCV     INVORDLIB   *FILE     PF          Parts receivable
 4      PARTSHP     INVORDLIB   *FILE     PF          Parts shipped
```

Working with Authorization Lists

When you have created the user profiles, you can create the authorization lists and then add users to them. The user profiles (or group profiles) must exist before users can be added to an authorization list.

If you completed a planning form (Authorization List Form (Part 1)), you can use the information on the completed forms when you create the authorization lists. If you did not complete a planning form, you can still create the authorization lists. However, you might want to write some notes about each list before you begin.

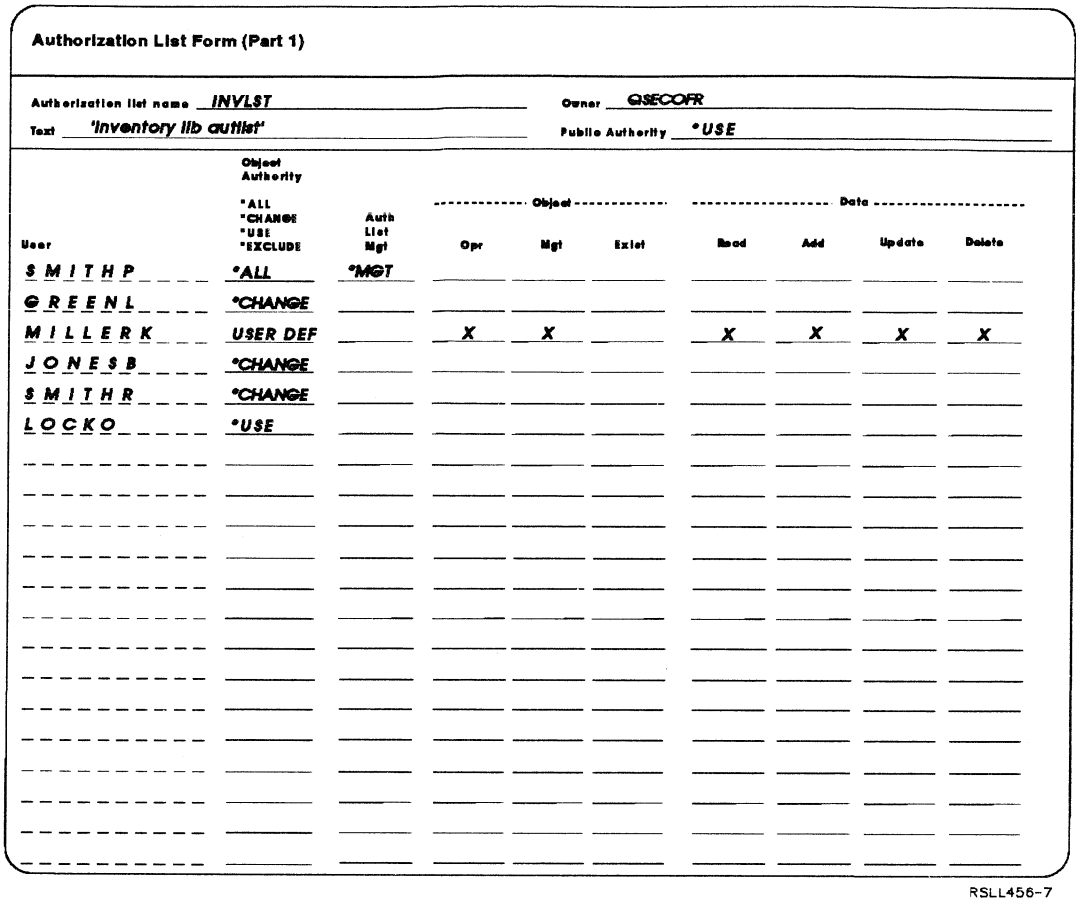
The following topics show step-by-step tasks to:

- Create an authorization list
- Add or remove users on an authorization list
- Grant or revoke an authorization list authority for an object
- Delete an authorization list

An object can be secured by only one authorization list.

Creating an Authorization List

If you have completed your planning, you should know how many authorization lists to create. You can use the information on the top part of the Authorization List Form (Part 1) when creating the authorization lists. Authorization lists are created using the Create Authorization List (CRTAUTL) command.



Authorization List Form (Part 1)

Authorization list name: INVLST
Owner: QSECOFR
Text: 'inventory lib autlist'
Public Authority: *USE

User	Object Authority *ALL *CHANGE *USE *EXCLUDE	Auth List Mgt	Object: Opr	Object: Mgt	Object: Exist	Data: Read	Data: Add	Data: Update	Data: Delete
SMITHP	*ALL	*MGT							
GREENL	*CHANGE								
MILLERK	USER DEF		X	X		X	X	X	X
JONESB	*CHANGE								
SMITHR	*CHANGE								
LOCKO	*USE								

RSLL456-7

Figure 8-27. Resource Security Form

1. To create an authorization list, type CRTAUTL on the command line of a menu, and press the F4 (Prompt) key.
2. Type the authorization list name for the *Authorization list* prompt and a description of the authorization list for the *Text 'description'* prompt. If you do not want to specify additional parameters, press the Enter key. If you do want to specify additional parameters, press the F10 (Additional parameters) key.

```
                Create Authorization List (CRTAUTL)

Type choices, press Enter.

Authorization list . . . . . .   invlst       Name
Text 'description' . . . . . .   'Inventory lib autlist'

                                                                    Bottom
F3=Exit   F4=Prompt  F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Figure 8-28. Create Authorization List Display

3. If you pressed F10 (Additional parameters), type the authority information for the *Authority* prompt, and press the Enter key.

```
                Create Authorization List (CRTAUTL)

Type choices, press Enter.

Authorization list . . . . . > invlst      Name
Text 'description' . . . . . > 'Inventory lib autlist'

                          Additional Parameters

Authority . . . . . .   *use        *CHANGE, *ALL, *USE, *EXCLUDE
```

Figure 8-29. Create Authorization List Display (Additional Parameters)

You can use the Create Authorization List (CRTAUTL) command and parameters as an alternative to the prompt displays. To bypass the prompt displays, you can enter the command and parameters on the command line of a menu, and press the Enter key.

```
CRTAUTL AUTL(INVLST) TEXT('Inventory lib autlist') AUT(*USE)
```

Adding and Removing Users on an Authorization List

After the authorization list is created, you can use the Edit Authorization List (EDTAUTL) command (interactive only) to add users to or remove users from the authorization list. You can also use the Add Authorization List Entry (ADDAUTLE) command (batch or interactive) to add a user to the authorization list or the Remove Authorization List Entry (RMVAUTLE) command (batch or interactive) to remove a user from the authorization list. In this example, the Edit Authorization List command is used.

1. Type EDTAUTL on the command line of a menu, and press the F4 (Prompt) key.

2. Type the name of the authorization list you want to add users to, and press the Enter key.

```
                    Edit Authorization List (EDTAUTL)

Type choices, press Enter.

Authorization list . . . . . .   invlst       Name
```

Figure 8-30. Edit Authorization List Display

The following display shows the name of the authorization list owner and the users (if any) on the list after F11 (Nondisplay detail) is pressed.

```
                         Edit Authorization List

Object . . . . . :   INVLST          Owner . . . . . :   QSECOFR
  Library  . . . . :   QSYS

Type changes to current authorities, press Enter.

             Object    List  ----Object-----  ----------Data-----------
User         Authority Mgt   Opr  Mgt  Exist  Read  Add  Update  Delete
QSECOFR      *ALL       X     X    X    X     X     X     X       X
*PUBLIC      *USE             X               X

                                                                    Bottom
F3=Exit F5=Refresh F6=Add new users   F11=Nondisplay detail  F12=Cancel
F15=Display auth list objects  F17=Top   F18=Bottom
```

Figure 8-31. Current Users on the Authorization List

3. To add or remove users from the authorization list, do one of the following:
   a. To add users to the list, press F6 (Add new users). Type the user profile name in the *User* column and the authority you are giving the user in the *Object Authority* column or in the *Object* and *Data* columns. Press the Enter key. After you press the Enter key, the users added to the list are shown on the display (see Figure 8-33 on page 8-35).
   b. To remove a user from the authorization list, replace the authority fields with blanks, and press the Enter key. After pressing the Enter key, the display shows that the user is removed from the list.

```
                              Add New Users

Object . . . . . :   INVLST          Owner . . . . . :   QSECOFR
  Library . . . . :     QSYS

Type new users, press Enter.

            Object    List  ----Object-----  ----------Data-----------
User       Authority  Mgt   Opr  Mgt  Exist  Read  Add  Update  Delete
smithp     *all        x
greenl     *change
millerk                     x    x          x     x      x   x
jonesb     *change
smithr     *change
locko      *use

                                                                   More...
F3=Exit   F11=Nondisplay detail   F12=Cancel   F17=Top   F18=Bottom
```

Figure 8-32. Adding Users to or Removing Users from the Authorization List

```
                          Edit Authorization List

Object . . . . . :   INVLST          Owner . . . . . :   QSECOFR
  Library . . . . :     QSYS

Type changes to current authorities, press Enter.

            Object    List  ----Object-----  ----------Data-----------
User       Authority  Mgt   Opr  Mgt  Exist  Read  Add  Update  Delete
QSECOFR    *ALL        X     X    X    X      X     X     X       X
GREENL     *CHANGE           X                X     X     X       X
JONESB     *CHANGE           X                X     X     X       X
LOCKO      *USE              X                X
MILLERK    USER DEF          X    X           X     X     X       X
SMITHP     *ALL        X     X    X    X      X     X     X       X
SMITHR     *CHANGE           X                X     X     X       X
*PUBLIC    *USE              X                X

                                                                   Bottom
F3=Exit  F5=Refresh F6=Add new users  F11=Nondisplay detail  F12=Cancel
F15=Display auth list objects  F17=Top  F18=Bottom
```

Figure 8-33. Edit Authorization List Display

You can use the Add Authorization List Entry (ADDAUTLE) or the Remove Authorization List Entry (RMVAUTLE) command and parameters as an alternative to the Edit Authorization List prompt displays. To bypass the prompt displays, type the command and parameters on a command line of a menu.

The following command adds a user to the authorization list:

```
ADDAUTLE AUTL(INVLST) USER(SMITHR) AUT(*ALL *AUTLMGT)
```

The following command removes a user from the authorization list:

```
RMVAUTLE AUTL(INVLST) USER(SMITHR)
```

Granting and Revoking an Authorization List Authority for an Object

After you have created the authorization list and added users to it, you can grant or revoke the authorization list authority for objects. If you completed Authorization List Form (Part 2), Resource Security for each authorization list, you can use the information on the completed form when you grant the authorization list authority for an object.

You can use the Edit Object Authority (EDTOBJAUT) command (interactive only) or the Grant Object Authority (GRTOBJAUT) command (batch or interactive) to grant and revoke an authorization list's authority to an object. In this example, the Edit Object Authority command is used.

1. Type EDTOBJAUT on the command line of a menu and press F4 (Prompt).
2. Type the name of the object for the *Object* prompt, the name of the library where the object is located for the *Library* prompt, and the type of object for the *Object type* prompt. Press the Enter key.

```
                Edit Object Authority (EDTOBJAUT)

Type choices, press Enter.

Object . . . . . .      invmnu     Name
  Library . . . . . .   invordlib  Name, *LIBL, *CURLIB
Object type . . . . .   *pgm      *ALRTBL, *AUTL, *CFGL...
```

Figure 8-34. Edit Object Authority Display

The following display shows the name of the object owner and the users who have authority to the object. If the object is secured by an authorization list, the authorization list name is shown.

```
                          Edit Object Authority

Object . . . . . :   INVMNU          Object type . . . . :   *PGM
  Library . . . . :    INVORDLIB     Owner . . . . . :   SMITHP

Type changes to current authorities, press Enter.

  Object secured by authorization list . . . . . .   *NONE

             Object    ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
SMITHP       *ALL        X    X     X     X     X     X       X
*PUBLIC      *CHANGE     X                X     X     X       X
                                                                  Bottom
F3=Exit  F5=Refresh  F6=Add new users  F10=Grant with reference object
F11=Nondisplay detail F12=Cancel       F17=Top  F18=Bottom
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Figure 8-35. Edit Object Authority Display

3. To grant an authorization list authority for the object, type the authorization list name for the *Object secured by authorization list* prompt, and press the Enter key.

```
                          Edit Object Authority

Object . . . . . :   INVMNU          Object type  . . . . :   *PGM
  Library  . . . . :     INVORDLIB     Owner  . . . . . . . :   SMITHP

Type changes to current authorities, press Enter.

  Object secured by authorization list  . . . . . . . . . . .   invlst

             Object    ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
SMITHP       *ALL        X    X     X     X     X     X       X
*PUBLIC      *CHANGE     X                X     X     X       X

                                                                  Bottom
 F3=Exit  F5=Refresh  F6=Add new users  F10=Grant with reference object
 F11=Nondisplay detail F12=Cancel       F17=Top  F18=Bottom
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Figure 8-36. Edit Object Authority Display

4. If you want the public authority for the object INVMNU to come from the authorization list, then change the authority for user *PUBLIC to *AUTL.

```
                          Edit Object Authority

Object . . . . . :   INVMNU          Object type  . . . . :   *PGM
  Library  . . . . :     INVORDLIB     Owner  . . . . . . . :   SMITHP

Type changes to current authorities, press Enter.
  Object secured by authorization list  . . . . . . . . . . .   invlst

             Object    ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
SMITHP       *ALL        X    X     X     X     X     X       X
*PUBLIC      *AUTL
```

Figure 8-37. Edit Object Authority Display

5. If you want to see who is on the authorization list that secures the object INVMNU, press F24 (More keys). Then press the F14 (Display auth list) key.
6. To revoke the authorization list's authority for the object, replace the authorization list name with *NONE, and press the Enter key.

You can use the Grant Object Authority (GRTOBJAUT) or the Revoke Object Authority (RVKOBJAUT) command and parameters as an alternative to the prompt displays. To bypass the prompt displays, you can type the command and parameters on the command line of a menu, and press the Enter key.

The following command grants the authorization list authority for the object:

```
GRTOBJAUT OBJ(INVORDLIB/INVMNU) OBJTYPE(*PGM) AUTL(INVLST)
```

The following command revokes the authorization list's authority for the object:

```
RVKOBJAUT OBJ(INVORDLIB/INVMNU) OBJTYPE(*PGM) AUTL(INVLST)
```

Displaying an Authorization List

You can use the Display Authorization List (DSPAUTL) command to see who is on an authorization list.

1. To display an authorization list, type DSPAUTL on the command line of a menu, and press the F4 (Prompt) key.
2. Type the name of the authorization list you want to display for the *Authorization list* prompt and the type of output you want for the *Output* prompt. Press the Enter key.

```
                  Display Authorization List  (DSPAUTL)

Type choices, press Enter.

Authorization list . . . . . .  invlst        Name
Output . . . . . . . . . . . .  *             *, *PRINT, *OUTFILE
```

After you press the Enter key and use F11 (Nondisplay detail), the following display is shown.

```
                          Display Authorization List

Object . . . . . :   INVLST          Owner. . . . . . . . :  QSECOFR
  Library  . . . . :     QSYS

              Object    List   ----Object-----  ----------Data-----------
User          Authority  Mgt    Opr  Mgt  Exist  Read  Add  Update  Delete
GREENL        *CHANGE           X                X     X     X       X
JONESB        *CHANGE           X                X     X     X       X
LOCKO         *USE              X                X
MILLERK       USER DEF          X    X           X     X     X       X
QSECOFR       *ALL         X    X    X     X     X     X     X       X
SMITHP        *ALL              X    X     X     X     X     X       X
SMITHR        *CHANGE      X    X    X     X     X     X     X       X
*PUBLIC       *EXCLUDE

                                                                    Bottom

F3=Exit   F11=Nondisplay detail   F12=Cancel
F15=Display authorization list objects  F17=Top   F18=Bottom
```

Figure 8-38. Display Authorization List Display

To bypass the prompt display, you can enter the following command and parameters on the command line of a menu, and press the Enter key:

```
DSPAUTL AUTL(INVLST)
```

Deleting an Authorization List

When an authorization list is no longer needed, it can be deleted even if users are still on the list. However, the authorization list cannot be deleted if it still secures objects. Authorization lists are deleted using the Delete Authorization List (DLTAUTL) command.

1. To delete an authorization list, type DLTAUTL on the command line of a menu, and press F4 (Prompt).
2. Type the name of the authorization list you want to delete for the *Authorization list* prompt, and press the Enter key.

```
                Delete Authorization List (DLTAUTL)

Type choices, press Enter.

Authorization list . . . . . .  invlst       Name, generic*
```

To bypass the prompt display, you can enter the following command and parameters on the command line of a menu, and press the Enter key:

```
DLTAUTL AUTL(INVLST)
```

Displaying Authority for Objects

You can display the specific authorities granted to users for an object. If you have object management authority for the specified object, if you are the owner of the object, or if you are the security officer, all users and their authorities are listed; otherwise, only your authority is listed. This list can be sent to an output file so it can be printed later.

If the group profile associated with the object has specific authorization for the object, the entry *GROUP is included in the list of users and its authority. This display can also be printed.

*ADOPTED is also displayed in the *User* column if a program (or programs in a program stack) is adopting the user's authority at the time of the display operation and has more authority than the user who is running the program that adopts.

The following example displays the specific authorities for the file INVMNU in the library INVORDLIB using the Display Object Authority (DSPOBJAUT) command.

1. To display the authority for an object, type DSPOBJAUT on the command line of a menu, and press the F4 (Prompt) key.
2. Type the name of the object for the *Object* prompt and the type of output you want for the *Output* prompt. Press the Enter key.

```
          Display Object Authority (DSPOBJAUT)

Type choices, press Enter.

Object . . . . . . . . . . . . .  invmnu       Name
  Library. . . . . . . . . . . .    invordlib  Name, *LIBL, *CURLIB
Object type. . . . . . . . . . .  *pgm         *ALRTBL, *AUTL, *CFGL. . .
Output . . . . . . . . . . . . .  *            *, *PRINT, *OUTFILE
```

Figure 8-39. Display Object Authority Display

After you press the Enter key, the following display is shown.

```
                        Display Object Authority

Object . . . . . . . :   INVMNU          Object type. . . . . :  *PGM
  Library  . . . . . :     INVORDLIB     Owner. . . . . . . . :  QSECOFR

Object secured by authorization list. . . . . . . . . . . . . :  INVLST

             Object     ----Object-----  ----------Data-----------
User         Authority  Opr  Mgt  Exist  Read  Add  Update  Delete
SMITHP       *ALL        X    X     X     X     X     X       X
*PUBLIC      *CHANGE     X                X     X     X       X
```

Figure 8-40. Display Object Authority Display

To bypass the prompt display, you can enter the following command from the command line of a menu, and press the Enter key:

```
DSPOBJAUT OBJ(INVORDLIB/INVMNU) OBJTYPE(*PGM)
```

The Display Object Authority display lists the name of a group profile (GROUP547) and the authority of the group if it has authority for the object. If you follow the recommended naming conventions for groups, you can identify the groups easily when you display the authorities for an object.

If you have signed on with your own user profile (for example, QPGMR) but are now running under another user profile, which has more authority than yours, a special object authority of *ADOPTED is shown on the Display Object Authority display. *ADOPTED represents the authority granted to the adopted user profile that is not granted to the original user profile.

If an object has been created with AUT(*EXCLUDE) and the owner removes his own authority to it, the owner sees the public authority and all authorities any user has been given specifically for the object.

If you enter the DSPOBJAUT command and an object without an owner is found, the *Owner* field on the display is set to blanks to indicate that the object has no owner. Only the public authority is shown.

The security officer or someone with all object (*ALLOBJ) special authority can use the Change Object Owner (CHGOBJOWN) command to assign an owner to an object that does not currently have an owner. An object with no owner occurs when the owner's user profile becomes damaged and is consequently deleted.

Security Consideration

Data authorities are not valid for logical files because the authority is controlled by the physical file. A user who is given *USE or *CHANGE authority for a logical file will have authority shown on the Display Object Authority display as USER DEF. The detailed authority on the Display Object Authority display only shows *OBJOPR because data authorities are not valid for logical files.

Consider the following when displaying authority for objects:

- If a group profile for an object is shown on the display, the members of the group also have authority for the object.

  To display the members of a group, use the Display User Profile (DSPUSRPRF) command, TYPE(*GRPMBR).
- If an authorization list is shown on the display, the users on the authorization list also have authority for the object. To see the users on the authorization list, press the F24 (More keys) key and then press the F14 (Display auth list) key or use the Display Authorization List (DSPAUTL) command. F14 (Display auth list) is not shown on the display if the object is not secured by an authorization list.

Displaying Programs That Adopt

The Display Program Adopt (DSPPGMADP) command shows all programs (and for Version 2 Release 1.1 SQL packages) that use the authority of the owning user profile. Through this command, the security officer, or someone with object management (*OBJMGT) authority for the object, can look at all programs (and for Version 2 Release 1.1 SQL packages) that adopt the owner's authority.

1. To display the programs (and for Version 2 Release 1.1 SQL packages) that adopt the owner's authority, type DSPPGMADP on the command line of a menu, and press the F4 (Prompt) key.
2. Type the name of the user profile for the *User profile* prompt and the type of output for the *Output* prompt. Press the Enter key.

```
                    Display Program Adopt (DSPPGMADP)

Type choices, press Enter.

User profile . . . . . .  kgreen      Name
Output . . . . . .  *                *, *PRINT, *OUTFILE
```

Figure 8-41. Display Programs That Adopt

After you press the Enter key, the following display is shown.

```
                        Display Programs That Adopt

User profile . . . . . :   KGREEN

Program     Library     Type      Attribute   Text
PAYROLL     MYLIB       *PGM      RPG         Payroll program
NEWEMP      PERSONNEL   *PGM      CLP         New employee education program

                                                                      Bottom
Press Enter to continue.

F3=Exit    F12=Cancel    F17=Top    F18=Bottom
(C) COPYRIGHT IBM CORP. 1980, 1991.
```

Figure 8-42. (For Version 2 Release 1.1) Display Programs That Adopt

To bypass the prompt display, you can enter the following command on the command line of any menu, and press the Enter key:

```
DSPPGMADP USRPRF(KGREEN)
```

The *CL Reference* manual contains additional information on the DSPPGMADP command.

Chapter 9. Security Questions and Answers

This chapter contains questions that are asked most often about security. Answers are provided for these questions.

QUESTION

Why did sign-on fail when the security officer tried to sign on a work station?

Answer

Users that have *ALLOBJ or *SERVICE special authorities must be given specific authority to the device. This problem can be solved in one of two ways:

- Grant the QSECOFR user profile *CHANGE authority to the device description.
- Change the QLMTSECOFR system value to '0' so the check for *ALLOBJ and *SERVICE is not done. See the topic "Display Station Considerations" on page 2-12 for more information about the QLMTSECOFR system value.

QUESTION

Why did the user profile not get adopted authority when running a program that is supposed to adopt the owner's authority?

Answer

If the program does not specify USRPRF(*OWNER) USEADPAUT(*YES), the program will not adopt the owner's authority when it is run.

Use the Display Program (DSPPGM) command to verify that the USRPRF parameter is specified as *OWNER and the USEADPAUT parameter is specified as *YES. If USRPRF(*OWNER) and USEADPAUT(*YES) are not specified, use the Change Program (CHGPGM) command to change the USRPRF parameter to *OWNER and the USEADPAUT parameter to *YES.

QUESTION

Why did user profiles lose special authorities when restored?

Answer

If user profiles are restored to a system at security level 10 or 20, and then the security level is changed to level 30 or above, the special authorities will be set to the defaults defined by the user class.

The Restore User Profile (RSTUSRPRF) command can be run at level 30 or above to restore special authorities for user profiles from tape. However, *ALLOBJ special authority is never restored for user profiles (except QSYS and QSECOFR) at security level 30 or above. If the user profile needs *ALLOBJ special authority, use the Change User Profile (CHGUSRPRF) command to change the special authority parameter in the user profile.

QUESTION

Why did authority holders suddenly appear on the system when I migrated from System/36?

Answer

When migrating from System/36, an authority holder is created for every file that is migrated. An authority holder is created for each entry in the resource security file that does not have a corresponding object on the system. If the authority holders are not needed, they can be deleted using the Delete Authority Holder (DLTAUTHLR) command.

QUESTION

Why was *ALLOBJ special authority removed when the user profile was restored?

Answer

When a user profile is restored to a system that has security level 30 or above, *ALLOBJ special authority is not restored (except to QSYS and QSECOFR user profiles). This is done

to prevent users that have only *SAVSYS special authority from restoring user profiles that have *ALLOBJ special authority. If the user profile needs *ALLOBJ special authority, use the Change User Profile (CHGUSRPRF) command to change the special authority parameter in the user profile.

QUESTION

Why does the user's group profile have authority to an object, but the user cannot access the object?

Answer

If the user profile has specific authority to the object, or the object is secured by an authorization list that the user is on, then the user's specific authority or authority specified on the authorization list will be used instead of the group profile authority. For more information about the order that authority is checked, see the topic "Authority Checking" on page 4-29.

To get the authority specified for the group, revoke the user's specific authority or remove the user from the authorization list.

QUESTION

When sign-on fails for a user profile because the job description (JOBD) or the output queue (OUTQ) associated with the user profile has been deleted, how do I correct the profile?

Answer

A job description and an output queue must exist before a user can sign on.

If the user profile is not the security officer (QSECOFR) user profile, then have the security officer change the JOBD and OUTQ parameters by using the Change User Profile (CHGUSRPRF) command. If the user profile is QSECOFR, call your service representative.

QUESTION

What happens to a user's private authorities when the user profile is restored?

Answer

Private authorities for objects are saved with the user profile when the Save System (SAVSYS) command or Save Security Data (SAVSECDTA) command is run. To restore private authorities for the objects that were restored, do the following:

1. Restore the user profiles by using the Restore User Profile (RSTUSRPRF) command.
2. Restore any objects that need to be restored by using the Restore Object (RSTOBJ) or Restore Library (RSTLIB) command.
3. Restore authority by using the Restore Authority (RSTAUT) command.

QUESTION

How do I get a list of all user passwords?

Answer

You cannot get a list of all user passwords on the system. Passwords are stored in a format that does not allow them to be displayed. If the security officer wants to be able to display passwords, consider the use of the DSPPWD tool in QUSRTOOL.

QUESTION

How can I get adopted authority in a submitted job?

Answer

See the topic "Submitting Jobs That Adopt Authority" on page 5-15 for a technique you can use when submitting jobs.

QUESTION

Where can I find what authority is needed to use a command?

Answer

Appendix D, "Authority Required for Objects Used by Commands" contains tables that list all commands that are not shipped with the public authority of *USE or require additional authority to the referenced object.

QUESTION

What can be done to prevent users from using trivial passwords?

Answer

Set guidelines for users so they do not use trivial passwords.

Use the following system values to control passwords:

Table 9-1. System Values That Apply to the Change Password Command

Value	Description
QPWDLMTAJC	Determines if digits can be next to each other in a new password
QPWDLMTCHR	Determines the characters that cannot be in a new password
QPWDLMTREP	Determines if repeating characters can be in a new password
QPWDMINLEN	Determines the minimum number of characters in a password
QPWDMAXLEN	Determines the maximum number of characters in a password
QPWDPOSDIF	Determines if each position in a new password must be different from the old password
QPWDRQDDGT	Determines if a digit is required in a new password
QPWDRQDDIF	Determines if the password must be different from the 32 previous passwords
QPWDVLDPGM	Specifies the name of the user-written password approval program

QUESTION

How do I prevent users from leaving their work stations signed on?

Answer

Use the QINACTITV system value to set the inactive time-out value. You can also restrict users to using one work station at a time by using the QLMTDEVSSN system value.

QUESTION

If I restrict users from signing on to more than one device, does that prevent them from using the system request menu or having multiple PC sessions?

Answer

Restricting a user to one device does not prevent them from accessing the system request menu. It does restrict a user from having multiple PC sessions on the same system.

QUESTION

Is there a way to recover if I lose the security officer's password, the DST password, or both?

Answer

If the security officer's password is lost, it can be reset to the IBM-supplied default using DST when an IPL is done. At the DST main menu, select the option to work with DST environment. At the Work with DST display, select the option to change the DST password. At the Change DST Password display, select the option to reset the security officer password.

If the DST password is lost, sign on as QSECOFR and use the Change DST Password (CHGDSTPWD) command to reset the DST password.

If both passwords are lost, contact your service representative.

QUESTION

Is there a way to restrict users from using the system request menu?

Answer

You can grant *EXCLUDE to the panel group QGMNSYSR for the user you want to restrict. Type the following command:

```
GRTOBJAUT QSYS/QGMNSYSR *PNLGRP user-name *EXCLUDE
```

If the user tries to use the System Request menu, a message is sent to the user indicating he is not authorized.

QUESTION

Why is giving a user *CHANGE authority to a source file not enough to do SEU operations?

Answer

SEU requires object management authority to add or change a member of the file. *CHANGE authority does not provide object management authority.

QUESTION

Is there a way to see all members of a group profile or all users on an authorization list?

Answer

To see all members associated with a group profile, use the Display User Profile (DSPUSRPRF) command on the group profile and specify *GRPMBR for the information type requested.

To see all users on an authorization list, use the Display Authorization List (DSPAUTL) command.

QUESTION

Is there a way to see all the objects that are secured by an authorization list?

Answer

To see a list of objects that are secured by the authorization list, use the Display Authorization List Objects (DSPAUTLOBJ) command.

QUESTION

If after the Restore Authority (RSTAUT) command is run, more libraries are found that need to be restored, how do I restore the private authorities?

Answer

To restore the private authorities, do the following:

1. Run the Restore User Profile (RSTUSRPRF) command to ensure that the owners of the libraries are on the system.
2. Restore the libraries.
3. Run the RSTAUT command to restore the private authorities for the libraries.

QUESTION

Why does a user who has the authority to an output queue receive a not-authorized exception when attempting to start a printer writer to the output queue?

Answer

If the authority to check (AUTCHK) parameter for the output queue is specified as *OWNER, then the user must either be the owner of the output queue or have *SPLCTL special authority to use the Start Printer Writer (STRPRTWTR) command on the output queue.

Use the Work with Output Queue Description (WRKOUTQD) command to determine the value for the AUTCHK parameter.

Appendix A. Security Commands

This appendix contains the system commands related to security. You can use these commands in place of the system menus, if you prefer, by typing these commands on a command line. The following commands are divided into task-oriented groups.

The *CL Reference* manual contains more detailed information about these commands.

Working with Authority Holders

1. CRTAUTHLR command

   The Create Authority Holder command allows you to secure an object before the associated object exists. Authority holders are only valid for program-described database files.

2. DLTAUTHLR command

   The Delete Authority Holder command allows you to delete an authority holder for an associated object.

3. DSPAUTHLR command

   The Display Authority Holder command allows you to display all the authority holders on the system.

Working with Authorization Lists

1. ADDAUTLE command

   The Add Authorization List Entry command allows you to add a user to an authorization list.

2. CHGAUTLE command

   The Change Authorization List Entry command allows you to change users' authorities in an authorization list.

3. CRTAUTL command

   The Create Authorization List command allows you to create an authorization list.

4. DLTAUTL command

   The Delete Authorization List command allows you to delete an entire authorization list.

5. DSPAUTL command

   The Display Authorization List command allows you to display a list of users assigned to an authorization list.

6. DSPAUTLOBJ command

   The Display Authorization List Objects command allows you to display a list of objects that use an authorization list.

7. EDTAUTL command

   The Edit Authorization List command allows you to add, change, and remove users and their authorities on an authorization list.

8. RMVAUTLE command

   The Remove Authorization List Entry command allows you to remove a user from an authorization list.

9. RTVAUTLE command

   The Retrieve Authorization List Entry command is used in a control language (CL) program to get one or more values associated with a user on the authorization list. The command can be used with the CHGAUTLE command to give a user new authorities in addition to the existing authorities that the user already has.

10. WRKAUTL command

    The Work with Authorization Lists command allows you to work with authorization lists from a list display.

Working with Object Authority

1. CHGOBJOWN command

   The Change Object Owner command allows you to change the ownership of an object from one user to another.

2. DSPOBJAUT command

   The Display Object Authority command allows you to display a list of users for an object and their associated authorities.

3. EDTOBJAUT command

   The Edit Object Authority command allows you to add, change, or remove a user's authority for an object.

4. GRTOBJAUT command

   The Grant Object Authority command allows you to specifically give authority to named users, all users (*PUBLIC), or users of the referenced object for the objects named in this command.

5. RVKOBJAUT command

   The Revoke Object Authority command allows you to remove one or more (or all) of the authorities given specifically to a user for the named objects.

6. WRKOBJ command

   The Work with Objects command allows you to work with object authority by selecting options on a list display.

7. WRKOBJOWN command

   The Work with Objects by Owner command allows you to work with the objects owned by any user profile.

Working with Passwords

1. CHGDSTPWD command

   The Change Dedicated Service Tools Password command allows you to reset the DST or the QSECOFR password to the default password shipped with the system.

2. CHGPWD command

   The Change Password command allows a user to change his password.

3. CHGUSRPRF command

   The Change User Profile command allows you to change the values specified in a user's profile such as the user's password special authorities, initial menu, initial program, current library, and priority limit.

4. CRTUSRPRF command

   The Create User Profile command allows you to add a user to the system and to specify values such as the user's password special authorities, initial menu, initial program, current library, and priority limit.

5. CHKPWD command

   The Check Password command allows a user to verify his password.

Working with User Profiles

1. CHGPRF command

   The Change Profile command allows a user to change some of the attributes of his user profile.

2. CHGUSRPRF command

   The Change User Profile command allows you to change the values specified in a user's profile such as the user's password special authorities, initial menu, initial program, current library, and priority limit.

3. CRTUSRPRF command

   The Create User Profile command allows you to add a user to the system and to specify values such as the user's password special authorities, initial menu, initial program, current library, and priority limit.

4. DLTUSRPRF command

   The Delete User Profile command allows you to delete a user profile from the system. This command provides an option to delete or change ownership of objects owned by the user profile.

5. DSPAUTUSR command

   The Display Authorized Users command allows you to display or print the names of authorized system users, the group profile name they are a member of, the date of the last time their passwords were changed, an indication of whether or not the user profiles have passwords, and the user profile text.

6. DSPPGMADP command

   The Display Programs Adopt command allows you to display a list of programs (and for Version 2 Release 1.1 SQL packages) that adopt a specified user profile.

7. DSPUSRPRF command

   The Display User Profile command allows you to display a user profile.

8. GRTUSRAUT command

   The Grant User Authority command allows you to give all the authority to a user profile by referencing another user profile.

9. RTVUSRPRF command

The Retrieve User Profile command is used in a control language (CL) program to get and use one or more values that are stored and associated with a user profile.

10. WRKUSRPRF command

    The Work with User Profiles command allows you to work with user profiles by entering options on a list display.

Related User Profile Commands

1. RSTAUT command

   The Restore Authority command allows you to restore authorities for objects held by a user profile when the user profile was saved. These authorities can only be restored after a user profile is restored with the Restore User Profile (RSTUSRPRF) command.

2. RSTUSRPRF command

   The Restore User Profile command allows you to restore a user profile and its attributes. Restoring specific authority to objects is done with the RSTAUT command after the restore of the user profile takes place. The RSTUSRPRF command also restores all authorization lists and authority holders if RSTUSRPRF(*ALL) is specified.

3. SAVSECDTA command

   The Save Security Data command saves all user profiles, authorization lists, authority holders, and mail objects on the system without using a system that is in a restricted state.

4. SAVSYS command

   The Save System command saves all user profiles, authorization lists, and authority holders on the system. A dedicated system is required to use this function.

Working with Document Library Objects

1. ADDDLOAUT command

   The Add Document Library Object Authority command gives a user access to a document or folder. This command allows you to specify authority for users in the following ways:

   - Grant specific authority to a user
   - Grant a set of users authority by specifying a previously defined authorization list
   - Grant a group of users access by specifying an access code

2. CHGDLOAUT command

   The Change Document Library Object Authority command is used to change an existing user's authority to a document or folder. This command allows the following changes in authority:

   - Change an existing user's specific authority
   - Change the authorization list specifying the object's security
   - Change public authority
   - Change the existing personal status of the DLO
   - Change the object's associated authorization list

3. CHGDLOOWN command

   The Change Document Library Object Owner command transfers document or folder ownership from one user to another user. The new owner must be in the system distribution directory. The authorities that other users have to the document are not changed. The user profile that no longer owns the object will have its current authority revoked unless the command user requests that the specific authorities be kept.

4. DSPAUTLDLO command

   The Display Authorization List Document Library Objects command is used to display the documents and folders that are secured by the specified authorization list.

5. DSPDLOAUT command

   The Display Document Library Object Authority command is used to display an existing user's authority to a document or folder. This command displays the following authority information for a document or folder:

   - Owner of the document or folder
   - Public authority
   - Personal status
   - Authorization list used to secure the document library object
   - Checkout status (document only)
   - Users' specific authority
   - Access code assigned to the document library object

6. EDTDLOAUT command

   The Edit Document Library Object Authority command is used to add, change, or remove user's authority to a document or folder. This command allows you to specify authority for users in the following ways:

   - Grant specific authority to a user
   - Add or remove an existing user's specific authority
   - Add, change, or remove the authorization list specifying the object's security
   - Change the personal status of the document library object
   - Reset the checkout status of the document
   - Change public authority
   - Add, change, or remove access codes

7. GRTUSRPMN command

   The Grant User Permission command gives permission to a user to handle documents and folders or to do office-related tasks on behalf of another user.

8. RMVDLOAUT command

   The Remove Document Library Object Authority command is used to remove an existing user's authority to documents or folders.

   The following types of authority can be removed:

   - An existing specific user's authority
   - The authorization list's authority for an object
   - An existing object access code

9. RVKUSRPMN command

   The Revoke User Permission command takes away document authority from one user (or all users) to access documents on behalf of another user.

Working with the System Distribution Directory

1. ADDDIRE command

   The Add Directory Entry command adds new entries to the system distribution directory. The directory contains information about a user, such as the user ID and address, system name, user profile name, mailing address, and telephone number. The ADDDIRE command provides a parameter for each of the fields contained in the directory.

2. CHGDIRE command

   The Change Directory Entry command changes the data for a specific entry in the system distribution directory. The system administrator has authority to update any of the data contained in a directory entry, except the user ID, address, and the user description. A user can only update his own directory entry and is limited to the fields that can be updated.

3. RMVDIRE command

   The Remove Directory Entry command removes a specific entry from the system distribution directory. When a user ID and address is removed from the directory, it is also removed from any distribution lists.

4. WRKDIR command

   The Work with Directory command provides a set of displays that allow a user to view, add, change, and remove entries in the system distribution directory. When the WRKDIR command is entered, the system shows one or all of the entries in the system distribution directory, depending on the parameters specified. If the parameter specified applies to more than one directory entry, the system displays a list of directory entries. If the parameter identifies a specific directory user, the system displays a list of entries for that user.

Appendix B. IBM-Supplied User Profiles

This appendix contains information about the user profiles that are shipped with the system.

Security Consideration

You should change the passwords of QSECOFR, QPGMR, QSYSOPR, QUSER, QSRV, and QSRVBAS as soon as you receive your system. These user profiles are the only IBM-supplied user profiles that a user can use to sign on.

In the following table, the first item, *Defaults*, lists the system-supplied defaults for each parameter of the IBM-supplied user profiles. If the parameter values are different for a specific IBM-supplied profile, those parameter values are noted for that profile. If the parameter uses the default value, then that parameter is not noted for that profile.

Table B-1 (Page 1 of 3). User Profiles

Object Name	Descriptive Name	Parameters
Defaults		Special Authority: *ALLOBJ [1] *SAVSYS [1] User Class: *USER Profile Status: *ENABLED Password Expired: *NO Assistance Level: *SYSVAL Current Library: *CRTDFT Initial Program: *NONE Initial Menu: MAIN Library: *LIBL Limited Capability: *NO Text: *BLANK Special Environment: *NONE Display Sign-On Information: *SYSVAL Password Expiration Interval: *SYSVAL Keyboard Buffering: *SYSVAL Limit Device Sessions: *SYSVAL Maximum Storage: *NOMAX Priority Limit: 0 Job Description: QDFTJOBD Library: *LIBL Group Profile: *NONE Owner: *USRPRF Group Authority: *NONE Accounting Code: *BLANK Document Password: *NONE Message Queue: *USRPRF Delivery: *NOTIFY Severity: 00 Print Device: *WRKSTN Output Queue: *WRKSTN Attention Program: *SYSVAL Language Identifier: *SYSVAL (for Version 2 Release 1.1) Country Identifier: *SYSVAL (for Version 2 Release 1.1) Coded Character Set Identifier: *SYSVAL (for Version 2 Release 1.1) User Option: *NONE Authority: *EXCLUDE

Table B-1 (Page 2 of 3). User Profiles

Object Name	Descriptive Name	Parameters
QSECOFR	Security officer user profile	Password: QSECOFR Special Authority: *ALLOBJ *SAVSYS *JOBCTL *SECADM *SPLCTL *SERVICE User Class: *SECOFR
QPGMR	Programmer user profile	Password: QPGMR Special Authority: *ALLOBJ [1] *SAVSYS *JOBCTL User Class: *PGMR Priority Limit: 3
QUSER	Work station user profile	Password: QUSER Maximum Storage: 3174K Priority Limit: 3
QSYSOPR	System operator user profile	Password: QSYSOPR Special Authority: *ALLOBJ [1] *SAVSYS *JOBCTL User Class: *SYSOPR Initial Menu: SYSTEM Library: *LIBL Maximum Storage: 6348K Message Queue: QSYSOPR Library: QSYS Delivery: *BREAK Severity: 40
QSRVBAS	Service basic user profile	Password: QSRVBAS Special Authority: *ALLOBJ [1] *SAVSYS [1] *JOBCTL Assistance Level: *INTERMED User Class: *PGMR Accounting Code: *SYS Attention Program: QSCATTN Library: QSYS
QSRV	Service user profile	Password: QSRV Special Authority: *ALLOBJ [1] *SAVSYS [1] *JOBCTL *SERVICE Assistance Level: *INTERMED User Class: *PGMR Accounting Code: *SYS Attention Program: QSCATTN Library: QSYS
QSPL	Spool user profile	Password: *NONE Accounting Code: *SYS
QSYS	System user profile	Password: *NONE Special Authority: *ALLOBJ *SECADM *SAVSYS *JOBCTL *SPLCTL *SERVICE User Class: *SECOFR Accounting Code: *SYS
QSPLJOB	Spool job user profile	Password: *NONE Accounting Code: *SYS
QRJE	Remote job entry user profile	Password: *NONE Special Authority: *ALLOBJ [1] *SAVSYS [1] *JOBCTL User Class: *PGMR
QDOC	Document user profile	Password: *NONE

Table B-1 (Page 3 of 3). User Profiles

Object Name	Descriptive Name	Parameters
QSNADS	Systems Network Architecture distribution services user profile	Password: *NONE Accounting Code: *SYS
QFNC	Finance user profile	Password: *NONE Priority Limit: 3 Accounting Code: *SYS
QDBSHR	Database share user profile	Password: *NONE Accounting Code: SYS
QTSTRQS	Test request user profile	Password: *NONE Accounting Code: *SYS
QGATE	VM/MVS* bridge user profile	Password: *NONE Accounting Code: *SYS
QDFTOWN	Default owner user profile	Password: *NONE Priority Limit: 3
QDSNX	Distributed systems node executive user profile	Password: *NONE Priority Limit: 3
QLPAUTO	Licensed program automatic install user profile	Password: *NONE Special Authority: *ALLOBJ, *JOBCTL, *SAVSYS *SECADM User Class: *SYSOPR Initial Program: QLPINATO Library: QSYS Initial Menu: *SIGNOFF Delivery: *BREAK Severity: 95
QLPINSTALL	Licensed program install user profile	Password: *NONE Special Authority: *ALLOBJ, *JOBCTL, *SAVSYS *SECADM User Class: *SYSOPR Delivery: *HOLD

Notes:

1 When the system security level is changed from level 10 or 20 to a level 30 or above, this value is removed.

Appendix C. Default Command Authorities of System-Supplied User Profiles

The Table C-1 on page C-2 identifies which IBM-supplied user profiles are authorized to use restricted commands. This table shows the restricted command authorizations that exist when the system is shipped.

The CL commands are listed in alphabetical order in the user profile table. The IBM-supplied user profiles are listed by their user-profile names across the top of the table.

The following list describes briefly the user profiles that are shipped with the system. (It does not include internal system profiles that are not intended for your use.)

User Profile	Description
QSECOFR	The security officer user profile has complete authority for all the CL commands. By using the security commands, the security officer can grant special authority to other users or revoke it. (The special authorities are described on the SPCAUT parameter of the Create User Profile (CRTUSRPRF) command description.)
QPGMR	The programmer user profile has authority for the commands that are normally used by a system or application programmer. The QPGMR user profile is authorized (when the system is shipped) to use most of the commands, except for those used only by the system operator or service personnel and those *restricted* to the security officer.
QSYSOPR	The system operator user profile has authority for the commands that are normally used by the system operator. The QSYSOPR user profile is authorized to use those commands that control the operation of the system, control the jobs that are active in the system, and do save and restore operations. When the system is shipped, QSYSOPR has the job control authority and save system authority assigned to it.
QSRV	The service user profile has authority for the commands using the service profile.
QSRVBAS	The service basic user profile has authority for the commands using the service basic service profile.
QDFTOWN	The system default owner user profile is given ownership if the previous owner of an object is no longer on the system.

The security officer can change the authority to commands for any of the IBM-supplied user profiles. He controls which commands are public and which users can use a command. Each command can be specifically authorized for one or more users. Some authority is usually needed for the OS/400 objects affected by the commands, as well as for the commands themselves.

Table C-1 on page C-2 shows the commands that are authorized for specific user profiles (indicated by an **S** under the profile name for which they are

authorized), and those that are restricted to the security officer only (indicated by an **R** in the QSECOFR profile column).

Cryptographic commands are shipped with only QSECOFR authority. All other commands not listed are public, which means they can be used by all users.

The security officer can use this table to change the user profile authorities for those commands listed and to indicate the changes on the table.

Table C-1 (Page 1 of 2). Authorities of IBM-Supplied User Profiles for Commands

Command Name	QPGMR (S)	QSYSOPR (S)	QSRV (S)	QSRVBAS (S)	QSECOFR (R)
ADDACC					R
ADDNETJOBE					R
ADDRPYLE	S				
ANSQST					R
ANZPRB	S	S	S	S	
APYJRNCHG	S		S		
APYPTF	S	S	S	S	
CFGDSTSRV	S	S			
CHGJRN	S	S	S		
CHGNETA					R
CHGNETJOBE					R
CHGPRB	S	S	S	S	
CHGPTR			S		
CHGQSTDB					R
CHGRPYLE	S				
CHGSYSLIBL					R
CHGSYSVAL	S	S	S		
CPYPTF	S	S	S	S	
CRTAPAR	S	S	S	S	
CRTAUTHLR					R
CRTQSTDB					R
CRTQSTLOD					R
DLTLICPGM					R
DLTQST					R
DLTQSTDB					R
DLTPRB	S	S	S	S	
DMPDLO	S	S	S	S	
DMPJOB	S	S	S	S	
DMPJOBINT	S	S	S	S	
DMPOBJ	S	S	S	S	
DMPSYSOBJ	S	S	S	S	
DSPDSTLOG					R
DSPPTF	S	S	S	S	
DSPSRVSTS	S	S	S	S	
EDTRBDAP		S			
EDTQST					R
ENDJOBABN	S	S	S		
ENDSRVJOB	S	S	S	S	
GRTACCAUT				S	R
HLDCMNDEV	S	S	S		R
HLDDSTQ	S	S		S	

Table C-1 (Page 2 of 2). Authorities of IBM-Supplied User Profiles for Commands

Command Name	QPGMR (S)	QSYSOPR (S)	QSRV (S)	QSRVBAS (S)	QSECOFR (R)
LODPTF	S	S	S	S	
LODQSTDB					R
PRTDOC	S	S	S	S	
PRTERRLOG	S	S	S	S	
PRTINTDTA	S	S	S	S	
RCLSPLSTG	S	S	S	S	
RCLSTG	S	S	S	S	
RCLTMPSTG	S	S	S	S	
RLSCMNDEV	S	S	S	S	
RLSDSTQ	S	S			S
RLSRMTPH	S	S			
RMVACC			S		
RMVACTTRA			S		
RMVJRNCHG	S		S		
RMVNETJOBE					R
RMVPTF	S	S	S	S	
RMVRPYLE	S				
RSTAUT					R
RSTCFG					R
RSTLICPGM					R
RSTUSRPRF					R
SAVLICPGM					R
SBMFNCJOB					R
SNDDSTQ	S	S			
SNDPTFORD			S		
SNDSRVRQS			S	S	
STRSST			S	S	
STRDBG	S		S		
STRSRVJOB	S	S	S	S	
TRCINT	S		S		
TRCJOB	S		S		
VFYCMN	S	S	S	S	
VFYLNKLPDA	S	S	S	S	
VFYPRT	S		S	S	
VFYTAP	S	S	S	S	
WRKCNTINF			S	S	
WRKDEVTBL					
WRKDPCQ	S	S			
WRKDSTQ	S	S			
WRKJRN	S	S	S		
WRKPGMTBL					R
WRKPRB	S	S	S	S	
WRKUSRTBL					R
WRKORDINF	S		S		
WRKSRVPVD			S	S	
WRKSRVRQS			S	S	

Appendix D. Authority Required for Objects Used by Commands

The tables in this appendix show what authority is needed for objects referenced by commands. This authority is required to do operations on the referenced objects. The tables are organized in alphabetical order according to object type. In addition, tables are included for items that are not OS/400 objects (jobs, spooled files, network attributes, and system values) and for some functions (device emulation and finance). Additional considerations (if any) for the commands are listed under the *Notes* section of each table.

The tables do not include all OS/400 commands. Commands that are shipped with the public authority of *USE (operational authority and read authority) and that do not require additional authority to referenced objects (*Referenced Object* column) are not included.

Referenced Object

The objects listed in the *Referenced Object* column are objects to which the user needs authority to use the command on that object.

Authority Needed

The authorities specified in the tables show the object authorities and/or the data authorities required for the object to use that command on the object.

The following is a brief description of the authorities that are specified in the *Authority Needed* column.

Operational, known as object operational authority, allows a user to look at the description of an object and use an object as determined by the data authorities that the user has for the object.

Management, known as object management authority, allows a user to specify the security for an object, move or rename an object, and add members to database files.

Existence, known as object existence authority, allows a user to control the existence and ownership of an object.

Authorization list management authority allows a user to add and remove users and their authorities on an authorization list.

Read authority allows a user to display the contents of an object or run a program.

Add authority allows a user to add entries to an object.

Update authority allows a user to change the entries in an object.

Change authority allows a user to change the contents of an object.

Use authority allows a user to create an object or display the contents of an object. Use authority allows the user to refer to the contents of an attached object when a command being requested must access attached objects and their contents.

Also, the *Authority Needed* column may be specified as a system-defined authority: *ALL, *CHANGE, *USE, or *EXCLUDE. These authorities are a subset of the object authorities and data authorities for an object. This subset combines one or more object authorities with one or more data authorities (see -- Table 'C1' unknown --). Authorization list management authority may be specified with one of the system-defined authorities on an authorization list. The only exception is exclude authority. If *EXCLUDE authority is specified, no other authority can be specified.

The following table shows the subset of object authorities and data authorities.

Table D-1. System-Defined Authority

	Object			Data			
Authority	***OBJOPR**	***OBJMGT**	***OBJEXIST**	***READ**	***ADD**	***UPD**	***DLT**
All	X	X	X	X	X	X	X
Change	X			X	X	X	X
Use	X			X			
Exclude	No	authority					

For more information on these authorities and their descriptions, see Chapter 4, "Resource Security."

The following information is assumed:

1. The term *security officer* means someone with all object (*ALLOBJ) special authority.
2. For commands, authority to the command is required.
3. To enter any display command, you need operational authority to the IBM-supplied display or printer output file used by the command.
4. To access any object, you need read authority to the library containing the object. This is specifically shown in the tables in this appendix.
5. To enter any create command, you need add authority to the library and to the user profile that becomes the owner of the created object.
6. To enter any delete command, you need delete authority to the library and to the user profile that owns the object.
7. Additional authority may be required to use specific functions called by the operation selected on the Work with (WRKxxxxx) commands. The user also needs authority for any commands called during the specific function.

Commands Common for All Objects

Command	Referenced Object	Authority Needed
ALCOBJ [1,2,3]	Allocated object	Operational
	Library	Read
CHGOBJD [4]	Object description	Management
	Library for object description	Read
CHGOBJOWN [3,4,5,6]	Object	Existence
	Library for object	Read
	Old user profile	Delete
	New user profile	Add
CHKOBJ [6]	Object	As requested by AUT keyword
	Library for object	Read
CPROBJ	Object	Management
	Library for object	Read
CRTDUPOBJ [4,7,16]	Object	Use and management
	To-library	Use and add
	From-library	Use
	If the object is an authorization list	Authorization list management
	For logical files	Operational and management
DCPOBJ	Object	Use
	Library for object	Read
DLCOBJ [1]	Object	Operational
	Library for object	Read
DMPOBJ [6,8]	Object	Use
	Library for object	Read
	Library, program, and user profile	Read
DMPSYSOBJ [8]	Object	Use
	Library for object	Read
	Library, program, and user profile	Read
DSPOBJAUT [6,9,17]	Display or printer output file	Operational
	Library for object	Read
DSPOBJD [2,17]	Display or printer output file	Operational
	Library for object	Read
EDTOBJAUT [3,6,10,11]	Object	Management
	Library for object	Read
GRTOBJAUT [3,4,6,10,11]	Object	Management
	Library for object	Read
MOVOBJ [4,6]	Object	Operational and management
	From-library	Delete and read
	To-library	Add and read
RCLSTG [12]		
RCLTMPSTG	Object	Management
	Library for object	Read
RNMOBJ [3,6,13,17]	Object	Management
	Library	Read and update
	If the object is an authorization list	Authorization list management

Commands Common for All Objects

Command	Referenced Object	Authority Needed
RSTOBJ [4,6,14]	Object	Existence
	Message queues being restored to library where they already exist	Operational
	User profile owning objects being created	Add
	To-library	Add and read
	Library for saved object if VOL(*SAVVOL) is specified	Use
RTVOBJD	Object	Authority other than *EXCLUDE
	Library	Read
RVKOBJAUT [4,10]	Object	Management
	Library for object	Read
SAVCHGOBJ [4,15]	Object	Existence
	Library for object	Read
SAVOBJ [4,6,15]	Object	Existence
	Library for object	Read
SAVSTG [14]		
SAVSYS [14]		
WRKOBJ	Object	Operational
	Library for object	Read
WRKOBJLCK	Object	Management
	Library for object	Read

Commands Common for All Objects

Command	Referenced Object	Authority Needed
WRKOBJOWN	User profile	Management
	Library for objects when option requested on object	Read

Notes:

1 See the OBJTYPE keyword of the ALCOBJ command for the list of object types that can be allocated and deallocated.

2 Ownership or some authority is required to the object.

3 For files, libraries, and subsystem descriptions, object operational authority is required to the object.

4 This command cannot be used for documents or folders.

5 Only someone with all object (*ALLOBJ) and security administrator (*SECADM) authority can change the object owner for programs adopting their owner's user profile.

6 You should use the equivalent Document Library Object commands for documents and folders.

7 For save files, you must be authorized to the Create Save File (CRTSAVF) command.

8 The public is not authorized to this command. IBM-supplied used profiles that have authority to use this command are QPGMR, QSRV, QSRVBAS, QSYSOPR, and QSECOFR.

9 If you have object management authority or you are the security officer, all owners and their authorities are shown. If you do not have object management authority, only your authorities are listed.

10 You must be the owner or have object management authority and the authorities being granted or revoked.

11 You must be the owner or security officer to grant object management authority.

12 The user profile calling the RCLSTG command must have authority to the RCLSTG command or it must have *ALLOBJ authority specified in the user profile.

13 This command cannot be used for user profiles, controller descriptions, device descriptions, line descriptions, documents, document libraries, folders, journals, and journal receivers.

14 You must have save system (*SAVSYS) special authority to use this.

15 If you have save system (*SAVSYS) special authority, you can save and restore any object. When saving to tape or diskette, you must have object operational authority to the device description and the device file for the tape or diskette. When saving to a save file, you must have object operational and add authorities to the save file. When restoring from a save file, you must have object operational and read authorities to the save file.

16 If ownership of the new object is changed to the group profile of the user running this command, (OWNER(*GRPPRF) specified in the user profile of the user running the command), the following restriction applies:

The user running this command must have private authority to the from-object. This does not include private authorities that are adopted. The user's private authority must be greater than or equal to all private authorities for any other users on the from-object. Otherwise, the private authority for the other user cannot be copied to the duplicated object.

Also, if another user has private authority to the from-object that includes *OBJMGT authority, this private authority *may not* be able to be copied to the new object. Only the owner of the object or a user with *ALLOBJ special authority can grant *OBJMGT authority. The user running the CRTDUPOBJ command can obtain the authority to the from-object through program adoption.

17 If OUTPUT(*OUTFILE) is specified, then the authority required by the CLRPFM and ADDPFM commands is also required.

Advanced Function Printing

Command	Referenced Object	Authority Needed
CRTFNTRSC	Source file	Use
	Library for font resource	Read and add
CRTFORMDF	Source file	Use
	Library for form definition	Read and add
CRTPAGSEG	Source file	Use
	Library for page segment	Read and add

Advanced Function Printing

Command	Referenced Object	Authority Needed
CRTPAGDFN	Source file	Use
	Library for page definition	Read and add
CRTOVL	Source file	Use
	Library for overlay	Read and add
DLTFNTRSC	Font resource	Existence
	Library for font resource	Read
DLTFORMDF	Form definition	Existence
	Library for form definition	Read
DLTPAGSEG	Page segment	Existence
	Library for page segment	Read
DLTPAGDFN	Page definition	Existence
	Library for page definition	Read
DLTOVL	Overlay	Existence
	Library for overlay	Read
WRKFNTRSC [1]	Font resource	Operational
	Library for font resource	Use
WRKFORMDF [1]	Form definition	Operational
	Library for form definition	Use
WRKPAGSEG [1]	Page segment	Operational
	Library for page segment	Use
WRKPAGDFN	Page definition	Operational
	Library for page definition	Use
WRKOVL [1]	Overlay	Operational
	Library for overlay	Read

Note:

[1] Ownership or some authority to the object is required.

Alert Description

Command	Referenced Object	Authority Needed
ADDALRD	Alert table	Use and add
	Library for alert table	Read
CHGALRD	Alert table	Use, add, and delete
	Library for alert table	Read
RMVALRD	Alert table	Operational and delete
	Library for alert table	Read
WRKALRD	Alert table	Operational
	To add alert description	Use and add
	To change alert description	Use, add, and delete
	To remove alert description	Operational and delete
	Library for alert table	Read

Alert Table

Command	Referenced Object	Authority Needed
CRTALRTBL	Library for alert table	Read and add
CHGALRTBL	Alert table	Change
	Library for alert table	Read
DLTALRTBL	Alert table	Existence
	Library for alert table	Read
WRKALRTBL	Alert table	Operational
	Library for alert table	Read

Alerts

Command	Referenced Object	Authority Needed
DLTALR	Physical file QAALERT	Change
WRKALR	Physical file QAALERT	Use

AS/400 CSP/AE

Command	Referenced Object	Authority Needed
CRTCSPAPP	From-file	Use
	To-file	Change
	DDS source file	Change and management
	Library	Read
CRTCSPMSGF	Message file	Change
	From-file	Use
CHGCSPPGM	Program	Change and management
DLTCSPMAP	Map group	Existence
DLTCSPTBL	Table	Existence

Authority Holder

Command	Referenced Object	Authority Needed
CRTAUTHLR	Associated object if it exists	All
DLTAUTHLR[1]	Authority holder	All
DSPAUTHLR[2]		

Notes:

1 If the associated file is a logical file, then object operational, object management, and object existence authority is required.

2 If OUTPUT(*OUTFILE) is requested, then the authority required by the Clear Physical File Member (CLRPFM) and the Add Physical File Member (ADDPFM) commands is also required.

Authorization List

Command	Referenced Object	Authority Needed
ADDAUTLE [1]	Authorization list	Authorization list management

Authorization List

Command	Referenced Object	Authority Needed
CHGAUTLE [1]	Authorization list	Authorization list management
DLTAUTL [2]		
DSPAUTL [3,5]	Authorization list	
DSPAUTLDLO	Authorization list	Use
DSPAUTLOBJ [3,5]	Authorization list	
EDTAUTL [1]		
RMVAUTLE [1]		
RTVAUTLE [1]		
WRKAUTL [4]	Authorization list	Operational
	Library QSYS	Read

Notes:

1 You must be the owner or have authorization list management authority and have the authorities being given, taken away, or retrieved.

2 You must be the owner or have all object (*ALLOBJ) special authority.

3 You must not be excluded (*EXCLUDE) from the list.

4 Ownership or some authority to the object is required.

5 If OUTPUT(*OUTFILE) is requested, then the authority required by the Clear Physical File Member (CLRPFM) and the Add Physical File Member (ADDPFM) commands is also required.

Chart

Command	Referenced Object	Authority Needed
DLTCHTFMT	Chart format	Existence
	Library for chart format	Read
DSPCHT	Chart format	Use
	Library for chart format	Use
	Database file	Use
	Library for chart format	Use
DSPGDF	Database file	Use
	Library for database file	Use
STRBGU [2] Option 3	Chart format	Change and existence
	Library for chart format	Use
WRKCHTFMT [1]	Chart format	Operational
	Library	Read

Notes:

1 Ownership or some authority to the object is required.

2 Option 3 on the BGU menu (shown when STRGBU is run) is the Change chart format option.

Class

Command	Referenced Object	Authority Needed
CRTCLS	Library for object	Read and add
DLTCLS	Object	Existence
	Library for object	Read

Class

Command	Referenced Object	Authority Needed
DSPCLS	Object	Operational
	Library for object	Read
WRKCLS [1]	Class-of-service description	Operational
	Library	Read

Notes:

1 Ownership or some authority to the object is required.

Class-of-Service Description

Command	Referenced Object	Authority Needed
CHGCOSD	Class-of-service description	Change
DLTCOSD	Class-of-service description	Operational and existence
DSPCOSD	Class-of-service description	Use
WRKCOSD [1,2]	Class-of-service description	Operational
	Library	Use

Notes:

1 To use individual operations, you must have the authority required by the individual operation.

2 Ownership or some authority to the object is required.

Commands

Command	Referenced Object	Authority Needed
CHGCMD	Command	Management
	Library for command	Read
CHGCMDDFT	Command	Management
	Library for command	Read
CRTCMD	Source file	Use
	Library for source file	Read and add
DLTCMD	Command	Existence
	Library for command	Read
DSPCMD	Command	Use
	Library for command	Read
PRTCMDUSG	Command	Operational
	Library for command	Read
SBMRMTCMD	Command	Operational
	Library for command	Read
	DDM file	Read
	Library for DDM file	Read
SLTCMD [1]	Command	Operational
	Library for command	Read

Commands

Command	Referenced Object	Authority Needed
WRKCMD [2]	Command	Operational
	Library for command	Read

Notes:

1 Ownership or some authority to the object is required.

2 To use the individual operations, you must have the authority required by the individual operation.

Configuration

Command	Referenced Object	Authority Needed
RSTCFG [1]	Object	Existence
	Message queues being restored to library where they already exist	Operational and existence
	To-library	Add and read
	User profile owning objects being created	Add
PRTDEVADR	Controller description	Use
	Device description	Use
RTVCFGSTS	Object	Operational
RTVCFGSRC	Object	Use
VRYCFG [2]	Object	Use
WRKCFGSTS [3]	Object	Operational

Notes:

1 Only the security officer can grant you authority to use this command.

2 Job control (*JOBCTL) special authority is also required.

3 To use the individual operations, you must have the authority required by the individual operation.

Configuration List

Command	Referenced Object	Authority Needed
ADDCFGLE	Configuration list	Change
CHGCFGL	Configuration list	Change
CPYCFGL	Configuration list	Use
CRTCFGL	Library for configuration list	Change
DLTCFGL	Configuration list	Operational and existence
DSPCFGL	Configuration list	Use
RMVCFGLE	Configuration list	Change
WRKCFGL [1]	Configuration list	Operational

Note:

1 To use the individual operations, you must have the authority required by the individual operation.

Connection List

Command	Referenced Object	Authority Needed
ADDCNNLE	Connection list	Change
CHGCNNL	Connection list	Change
CRTCNNL	Connection list	Use
DLTCNNL	Connection list	Operational and existence
DSPCNNL	Connection list	Use
RMVCNNLE	Connection list	Change
RNMCNNLE	Connection list	Change
WRKCNNL [1]	Connection list	Operational
WRKCNNLE [1]	Connection list	Operational

Note:

[1] To use the individual operations, you must have the authority required by the individual operation.

Controller Descriptions

Command	Referenced Object	Authority Needed
CHGCTLAPPC	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
	Connection list specified in the CNNLSTOUT parameter	Use
CHGCTLASC	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
CHGCTLBSC	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
CHGCTLFNC	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
CHGCTLHOST	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
CHGCTLLWS	Controller description	Change
CHGCTLNET	Controller description	Change
CHGCTLRTL	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
CHGCTLRWS	Controller description	Change
	Line description specified in the SWTLINLST parameter	Use
	Connection list specified in the CNNLSTOUT parameter	Use
CHGCTLTAP	Controller description	Change
CHGCTLVWS	Controller	Change

Controller Descriptions

Command	Referenced Object	Authority Needed
CRTCTLAPPC	Line description specified in the LINE and/or SWTLINLST parameters	Use
	Device description specified in the DEV parameter	Use
	Connection list specified in the CNNLSTOUT parameter	Use
CRTCTLASC	Line description specified in the LINE or SWTLINLST parameter	Use
	Device description specified in the DEV parameter	Use
CRTCTLBSC	Line description specified in the LINE or SWTLINLST parameter	Use
	Device description specified in the DEV parameter	Use
CRTCTLFNC	Line description specified in the LINE or SWTLINLST parameter	Use
	Device description specified in the DEV parameter	Use
CRTCTLHOST	Line description specified in the LINE or SWTLINLST parameter	Use
	Device description specified in the DEV parameter	Use
	Connection list specified in the CNNLSTOUT parameter	Use
CRTCTLLWS	Device description specified in the DEV parameter	Use
CRTCTLNET	Line description specified in the LINE parameter	Use
	Device description specified in the DEV parameter	Use
CRTCTLRTL	Line description specified in the LINE or SWTLINLST parameter	Use
	Device description specified in the DEV parameter	Use
CRTCTLRWS	Line description specified in the LINE or SWTLINLST parameter	Use
	Device description specified in the DEV parameter	Use
	Connection list specified in the CNNLSTOUT parameter	Use
CRTCTLTAP	Device description specified in the DEV parameter	Use
CRTCTLVWS	Device description specified in the DEV parameter	Use
DLTCTLD	Controller description	Operational and existence
DSPCTLD	Controller description	Use
ENDCTLRCY	Controller description	Operational
RSMCTLRCY	Controller description	Operational
VFYCMN [1]	Object	Use

Controller Descriptions

Command	Referenced Object	Authority Needed
WRKCTLD [2]	Controller description	Operational

Notes:

1 Only QSYSOPR, QPGMR, QSRV, QSRVBAS, or QSECOFR user profiles can use this command

2 To use the individual operations, you must have the authority required by the individual operation.

Cryptography

Command	Referenced Object	Authority Needed
ADDCRSDMNK	Physical file QACRKTBL	Use and add
CHGCRSDMNK	Physical file QACRKTBL	Use and update
CHGMSTK	Physical file QACRKTBL	Use and update
ENCFRMMSTK	Physical file QACRKTBL	Use
ENCTOMSTK	Physical file QACRKTBL	Use
GENCPHK	Physical file QACRKTBL	Use
GENCRSDMNK	Physical file QACRKTBL	Use and add
GENPIN	Physical file QACRKTBL	Use
RMVCRSDMNK	Physical file QACRKTBL	Use and delete
SETMSTK	Physical file QACRKTBL	Use and update
TRNPIN	Physical file QACRKTBL	Use
VFYPIN	Physical file QACRKTBL	Use

Data Areas

Command	Referenced Object	Authority Needed
CHGDTAARA [1]	Data area	Operational and update
	Library for data area	Read
CRTDTAARA [1]	Library for data area	Read and add
DLTDTAARA	Data area	Existence
	Library for data area	Read
DSPDTAARA	Data area	Operational
	Library for data area	Read
RTVDTAARA [2]	Data area	Operational
	Library for data area	Read
WRKDTAARA [3]	Data area	Operational
	Library for data area	Read

Notes:

1 If the create and change data area commands are run using high-level language functions, these authorities are still required although authority to the command is not.

2 Authority is verified at run time, but not at compilation time.

3 Ownership or some authority to the object is required.

Data Queues

Command	Referenced Object	Authority Needed
CRTDTAQ	Library for data queue	Add and read
	Data queue to send the data queue using the QSNDDTAQ program	Operational and add
	Data queue to receive the data queue using the QRCVDTAQ program	Use
DLTDTAQ	Data queues	Existence
	Library for data queue	Read
WRKDTAQ [1,2]	Data queues	Operational
	Library for data queue	Use

Notes:

1 To use individual operations, you must have the authority required by the individual operation.

2 Ownership or some authority to the object is required.

Device Descriptions

Command	Referenced Object	Authority Needed
CHGDEVAPPC	Device description	Change
	Mode description specified in the MODE parameter	Use
CHGDEVASC	Device description	Change
CHGDEVBSC	Device description	Change
CHGDEVDKT	Device description	Change
CHGDEVDSP [2]	Device description	Change
	Printer specified in the PRINTER parameter	Use
CHGDEVFNC	Device description	Change
CHGDEVHOST	Device description	Change
CHGDEVINTR	Device description	Change
CHGDEVPRT	Device description	Change
CHGDEVRTL	Device description	Change
CHGDEVSNUF	Device description	Change
CHGDEVTAP	Device description	Change
CRTDEVAPPC	Controller description specified in the CTL parameter	Use
CRTDEVASC	Controller description specified in the CTL parameter	Use
CRTDEVBSC	Controller description specified in the CTL parameter	Use
CRTDEVDSP	Printer description specified in the PRINTER parameter	Use
	Controller description specified in the CTL parameter	Use
CRTDEVFNC	Controller description specified in the CTL parameter	Use
CRTDEVHOST	Controller description specified in the CTL parameter	Use
CRTDEVPRT	Controller description specified in the CTL parameter	Use

Device Descriptions

Command	Referenced Object	Authority Needed
CRTDEVRTL	Controller description specified in the CTL parameter	Use
CRTDEVSNUF	Controller description specified in the CTL parameter	Use
DLTDEVD [1]	Device description	Operational and existence
DSPCNNSTS	Device description	Operational
DSPDEVD	Device description	Use
ENDDEVRCY	Device description	Operational
HLDCMNDEV [2]	Device description	Operational
RLSCMNDEV	Device description	Operational
RSMDEVRCY	Device description	Operational
WRKDEVD [3]	Device description	Operational

Notes:

1 To remove an associated output queue, object existence (*OBJEXIST) authority to the output queue and read authority to the QUSRSYS library are required.

2 You must have job control (*JOBCTL) special authority and object operational authority to the device description.

3 To use individual operations, you must have the authority required by the individual operation.

Device Emulation

Command	Referenced Object	Authority Needed
EJTEMLOUT	Emulation device description when specified	Operational
	Printer device description when specified	Operational
	Emulation device description when location specified	Operational
ENDPRTEML	Emulation device description when specified	Operational
	Printer device description when specified	Operational
	Emulation device description when location specified	Operational
EMLPRTKEY	Emulation device description when specified	Operational
	Printer device description when specified	Operational
	Emulation device description when location specified	Operational
STREML3270	Emulation device, display station device, and display station controller descriptions	Operational
	Printer device description, emulation controller description, and translate tables when specified	
STRPRTEML	Emulation device description	Operational
	Printer device description, emulation controller description, print file, message queue, job description, and translate tables when specified	Operational
SNDEMLIGC	From-file	Operational

Directory

Command	Referenced Object	Authority Needed
ADDDIRE [1]		
CHGDIRE [2]		
RMVDIRE [1]		

Notes:

1 Only a user with security administrator (*SECADM) special authority can use this command.

2 A user can change his own directory entry. A user with *SECADM special authority can change any user's directory entry.

Display Station Pass-Through

Command	Referenced Object	Authority Needed
STRPASTHR	APPC device on source system	Use
	APPC device on target system	Change
	Virtual controller on target system [1]	Use
	Virtual device on remote system [1,2]	Change
	Program specified in the QRMTSIGN system value on remote system, if any[1]	Use
	Library for program specified in the QRMTSIGN system value on remote system, if any [1]	Use

Notes:

1 The user profile that requires this authority is the profile that runs the pass-through batch job. For pass-through that bypasses the sign-on display, the user profile is the one specified in the remote user (RMTUSER) parameter. For pass-through that uses the normal sign-on procedure (RMTUSER(* NONE)), the user is the default user profile specified in the communications entry of the subsystem that handles the pass-through request. Generally, this is QUSER.

2 If the pass-through is one that uses the normal sign-on procedure, the user profile specified on the sign-on display on the target system must have authority to this object.

Distribution

Command	Referenced Object	Authority Needed
CFGDSTSRV [1]		
CFGRPDS [1]		
CHGDSRQSTS [2]		
CHGDSTD	Document	Change
DLTDST [3]		
QRYDST [4]	Requested file	Change
RCVDST	Requested file	Change
	Folder	Change
SNDDST	Requested file or document	Use
WRKDSTQ [5]		

Distribution

Command	Referenced Object	Authority Needed
WRKDPCQ [5]		

Notes:

1 Only the QPGMR and QSECOFR user profiles can use this command.

2 Only the QSYSOPR user profile can use this command.

3 Only the owner of the distribution list or a user with security administrator (*SECADM) special authority can use this command.

4 If the user is asking for distribution other than his own, he must have the authority to work on behalf of another user to use this command.

5 Only the QSYSOPR, QPGMR, and QSECOFR user profiles can use this command.

Distribution List

Command	Referenced Object	Authority Needed
ADDDSTLE [1]		
DLTDSTL [1]		
RMVDSTLE [1]		

Note:

1 Only the owner of the distribution list or a user with security administrator (*SECADM) special authority can use this command.

Document Library Objects

Command	Referenced Object	Authority Needed
ADDDLOAUT [3]	Document library object	All
CHGDLOAUT [3]	Document library object	All
CHGDLOOWN [1,3]	Document library object	Owner or security administrator *SECADM
	Old user profile	Delete
	New user profile	Add
CHGDOCD [2,3]	Document description	Change
CHKDLO [2,3]	Document library object	As required by the AUT keyword
CRTFLR [3]	In-folder	Change
DLTDLO [3,8]	Document library object	All
DMPDLO [3]	Document library object	Use
DSPAUTLDLO [3]	Authorization list	Use
DSPDLOAUT [3]	Document library object	Use
DSPFLR [3]	Folder	Use
EDTDLOAUT [3]	Document library object	All
QRYDOCLIB [2,9]	Requested file	Use
RMVDLOAUT [3]	Document library object	All
RNMDLO [5]	Document library object	All
	In-folder	Change
RSTDLO [3,4,5]	Document library object	All
RSTS36FLR [4,6]	S/36 folder being restored	Use
	To-folder	Change

Document Library Objects

Command	Referenced Object	Authority Needed
SAVDLO [3,5,7]	Document library object	All

Notes:

1 You must be the owner or have security administrator (*SECADM) special authority.

2 If the user is working on behalf of another user, the other user's authority to the object is checked.

3 If you have security administrator (*SECADM) special authority or all object (*ALLOBJ) special authority, you do not need all (*ALL) authority to the document library object.

4 You need operational, management, existence, and all data authorities to the document if replacing it. You need operational and all the data authorities to the folder if restoring new information into the folders, or you need save system (*SAVSYS) special authority or all object (*ALLOBJ) special authority.

5 If you have save system (*SAVSYS) special authority or all object (*ALLOBJ) special authority, you can save and restore any document library object. You must have operational authority to the device and device file for save and restore media.

6 If used for a data dictionary, only the authority to the command is required.

7 Operational, management, existence, and all data authorities are required, or you must have *SAVSYS special authority or all object (*ALLOBJ) special authority.

8 The user must have *ALL authority to all the objects in the folder in order to delete them.

9 Only objects that meet the criteria of the query ***and*** to which the user has at least use (*USE) authority will be returned in the document list or outfile. Authority is not checked if user has all object (*ALLOBJ) special authority.

Documents

Command	Referenced Object	Authority Needed
CHKDOC	Document	Change
	Spelling aid dictionary	Change
CPYDOC	From-document	Use
	To-document if replaced	Change
	To-folder if to-document is new	Change
CRTDOC	In-folder	Change
DSPDOC	Document	Use
EDTDOC	Document	Change
FILDOC [1]	Requested file	Use
	Folder	Change
MOVDOC	From-folder [4]	Change
	From-document	Use
	To-folder [6]	Change
	To-document [5,6]	
MRGDOC [2]	Document	Use
	From-folder	Use
	To-document (if document is replaced)	Change
	To-folder (if to-document is new)	Change
PAGDOC	Folder	Change
	Document	Change
PRTDOC[3]	Folder	Use
	Document	Use
RPLDOC [1]	Requested file	Read
	Document	Change

Documents

Command	Referenced Object	Authority Needed
RTVDOC [1]	Document if checking out	Change
	Document if not checking out	Use
	Requested file	Change
WRKDOC	Folder	Use

Notes:

1 If the user is working on documents on behalf of another user, the other user's authority to the object is checked.

2 The user must have authority to the object being used as the merge source. For example, if MRGTYPE(*QRY) is specified, the user must have use authority to the query specified for the QRYDFN parameter.

3 If the document being printed contains an *Index* instruction, the user must have use authority to the following commands: CRTPF, DLTF, and DLTOVR.

If the document being printed contains a *Run* instruction, the user must have use authority to the following commands: CRTPF, OVRPRTF, DLTSPLF, and DLTOVR.

Depending on the parameters specified on the command, use or change authority to other objects may also be required. For example, if SAVOUTPUT(*YES) is specified, the user must have change authority to the document specified on the SAVDOC parameter and the folder specified on the SAVFLR parameter.

4 There can be documents in the document library that are not in a folder. These documents can be moved to a folder by specifying the object name of the from-document. In this case, only the user's authority to the document is checked.

5 If the to-document already exists in the to-folder, the user must have *CHANGE authority to the to-document being replaced. The user's authority to the to-folder is not checked.

6 If *NONE is specified for the to-folder, to-document is not allowed. Only the user's authority to the from-folder and from-document is checked.

Double-Byte Character Set

Command	Referenced Object	Authority Needed
CPYIGCTBL	Table for copy-in function	Operational
	Library for table if table does not exist for copy-in function	Add
	Library for table	Read
CRTIGCDCT	Library for DBCS conversion dictionary	Read and add
DLTIGCDCT	DBCS conversion dictionary	Existence
	Library for DBCS conversion dictionary	Read
DLTIGCSRT	DBCS sort table	Existence
	Library for sort table	Read
DLTIGCTBL	DBCS font table	Existence
DSPIGCDCT	DBCS conversion dictionary	Use
	Library for DBCS conversion dictionary	Read
EDTIGCDCT	DBCS conversion dictionary	Use and update
	User dictionary being edited	Add and delete to user dictionary being edited
STRCGU	DBCS sort table	Change
	DBCS font table	Change

Edit Descriptions

Command	Referenced Object	Authority Needed
CRTEDTD	Library QSYS	Read and add

Edit Descriptions

Command	Referenced Object	Authority Needed
DLTEDTD	Edit description	Existence
	Library QSYS	Read
DSPEDTD	Edit description	Operational
	Library QSYS	Read
WRKEDTD [1]	Edit description	Operational
	Library QSYS	Read

Note:

[1] Ownership or some authority to the object is required.

Files

Command	Referenced Object	Authority Needed
ADDICFDEVE	ICF file	Operational and management
	Library for ICF file	Read
ADDLFM	Logical file	Operational and management
	Files referenced in DTAMBRS parameter	Operational and management
	Library for logical file	Read and add
ADDPFM	Physical file	Operational and management
	Library for physical file	Read and add
CHGDDMF	DDM file	Operational and management
	Library for DDM file	Read
CHGDKTF	Diskette file	Operational and management
	Device if device name specified in the command	Operational
	Library for diskette file	Read
CHGDSPF	Display file	Operational and management
	Device if device name specified	Operational
	Library for display file	Read
CHGDTA	Data file	Operational, add, update, and delete
	Library for data file	Read
CHGICFDEVE	ICF file	Operational and management
	Device if device name specified	Operational
	Library for ICF file	Read
CHGICFF	ICF file	Operational and management
	Library for ICF file	Read and add
CHGLF	Logical file	Operational and management
	Library for logical file	Read
CHGLFM	Logical file	Operational and management
	Library for logical file	Read
CHGPF	Physical file	Operational and management
	Library for physical file	Read
CHGPFM	Physical file	Operational and management
	Library for physical file	Read

Files

Command	Referenced Object	Authority Needed
CHGPRTF	Print file	Operational and management
	Device if device name specified	Read
	Library for print file	Operational
CHGSAVF	Save file	Operational and management
	Library for save file	Read
CHGSRCPF	Source physical file	Operational and management
	Library for source physical file	Read
CHGTAPF	Tape file	Operational and management
	Device if device name specified	Operational
	Library for tape file	Read
CLRPFM	Physical file	Operational, management, and delete
	Library for physical file	Read
CLRSAVF	Save file	Operational and management
	Library	Read
CPYF	To-file and from-file	Operational
	To-file when a device file	Read
	To-file that is a physical file with MBROPT(*REPLACE) specified	Management, add, and delete
	To-file that is a physical file with MBROPT(*ADD) specified	Add
	From-file	Read
	Based-on file if from-file is logical file	Read
	Library for to-file	Read and add
	Library for from-file	Read
	CRTPF command if CRTFILE(*YES) is specified	Operational
CPYFRMDKT [1]	To-file and from-file	Operational
	To-file when a device file	Read
	To-file that is a physical file with MBROPT(*REPLACE) specified	Management, add, and delete
	To-file that is a physical file with MBROPT(*ADD) specified	Add
	Library for to-file	Read and add
	Library for from-file	Read
CPYFRMQRYF	To-file	Operational
	To-file when a device file	Read
	To-file that is a physical file with MBROPT(*REPLACE) specified	Management, add, and delete
	To-file that is a physical file with MBROPT(*ADD) specified	Add
	Library for to-file	Read and add
	CRTPF command if CRTFILE(*YES) is specified	Operational

Files

Command	Referenced Object	Authority Needed
CPYFRMTAP	To-file and from-file	Operational
	To-file when a device file	Read
	To-file that is a physical file with MBROPT(*REPLACE) specified	Management, add, and delete
	To-file that is a physical file with MBROPT(*ADD) specified	Add
	From-file	Read
	Library for to-file	Read
	Library for from-file	Read
CPYSRCF	To-file and from-file	Operational
	To-file if MBROPT(*REPLACE) specified	Operational, management, add and delete
	To-file if MBROPT(*ADD) specified	Add
	From-file	Read
	Library for to-file	Read and add
	Library for from-file	Read
CPYTODKT [1]	To-file and from-file	Operational and read
	Device if device name specified on the command	Operational
	Based-on physical file if from-file is logical file	Read
	Library for to-file	Read
	Library for from-file	Read
CPYTOTAP [1]	To-file and from file	Use
	Device if device name is specified	Operational
	Based-on physical file if from-file is logical file	Read
	Library for to-file	Read
	Library from file	Read
CRTDDMF	Library	Read and add
CRTDKTF	Device if device name is specified	Operational
	Library for diskette file	Read and add
CRTDSPF	Source file	Use
	Device if device name is specified	Operational
	File specified in REF and REFFLD keywords	Operational
	Library for display file	Read and add
CRTICFF	Source file	Use
	File specified in REF and REFFLD keywords	Operational
	Device file if device name is specified	Operational
	Library for source file	Read and add
	Library for device file	Read and add
CRTLF	Source file	Use
	Files specified in PFILE or JFILE keywords and file specified on REFACCPTH keyword	Operational and management
	Files specified in FORMAT and tables specified in the ALTSEQ keyword	Operational
	Library for logical file	Read and add

Files

Command	Referenced Object	Authority Needed
CRTPF	Source file	Use
	Files specified in FORMAT and REFFLD keywords and tables specified in the ALTSEQ keyword	Operational
	Library for physical file	Read and add
CRTPRTF	Source file	Use
	Device if device name is specified	Operational
	Library for print file	Read and add
	Files specified in the REF and REFFLD keywords	Operational
CRTSAVF	Library for save file	Read and add
CRTSRCPF	Library for source physical file	Read and add
CRTS36DSPF	To-file source file when TOMBR is not *NONE	All
	To-file library	Change
	Source file QS36SRC	Use
	Source file library	Read
	Display file when REPLACE(*YES) is specified.	All
	Display file library	Read and add
	Create Display File (CRTDSPF) command	Operational
CRTTAPF	Library for tape file	Read and add
	Device if device name is specified	Operational
DLTF	File	Operational and existence
	Library for file	Read
DLTQRY	Query definition	Existence
	Library for query definition	Read
DLTSCHIDX	Search index	Existence
	Library for index search	Read
DSPDBR	Database file	Operational
	File specified in OUTFILE parameter	Operational, management, add, and delete
	CRTPF command if CRTFILE(*YES) is specified	Operational
	Libraries for database files	Read
	Library specified on OUTFILE parameter if file does not exist	Add
	Library specified on OUTFILE parameter if file does exist	Read
DSPDDMF	DDM file	Operational
	Library for DDM files	Read
DSPDTA	Data file	Use
	Library for data file	Read

Files

Command	Referenced Object	Authority Needed
DSPFD [2]	Files	Operational
	File specified in OUTFILE parameter	Operational, management, add, and delete
	Library for files	Read
	File is a physical file and TYPE(*ALL, *MBR, OR *MBRLST) is specified	One data authority (read, add, update, or delete)
	Library specified on OUTFILE parameter if file does not exist	Add
	Library specified on OUTFILE parameter if file does exist	Read
DSPFFD	Files	Operational
	File specified in OUTFILE parameter	Operational, management, add, and delete
	CRTPF command if CRTFILE(*YES) is specified	Operational
	Library specified on OUTFILE parameter if file does not exist	Add
	Library specified on OUTFILE parameter if file does exist	Read
	Libraries for files	Read
DSPMSGF	Message file	Use
DSPPFM	Physical file	Use
	Library for physical file	Read
DSPSAVF	Save file	Use
	Library for save file	Read
ENDCMTCTL	Message queue	Add
	Data area	Update
	Library for files, message queue, and data area	Read
INZPFM	Physical file	Operational, management, and add
	Initialize when RECORD(*DLT) is specified	Delete
OPNDBF	Database file	Operational
	Library for database file	Read
OPNQRYF	Query file	Operational
	Library for query file	Read
RGZPFM	File	Change and management
	Library for file	Read
RMVICFDEVE	ICF file	Operational and management
	Library for ICF file	Operational
	Device if device name specified	Operational
	Library	Read
RMVM	File member	Operational and existence
	File containing member	Operational and delete
	Library for file	Read
RNMM	File containing member	Operational and management
	Library for file	Read and update

Files

Command	Referenced Object	Authority Needed
RSTS36F [4]	To-file to add members	Change and management
	To-file to replace existing member	All
	Based on physical file if file being restored is a logical (alternative) file	Change
	To-library if file being restored already exists	Use
	To-library if file being restored does not exist	Change
	Device description for diskette or tape	Use
	Library and file if to-file (and library) is a physical file	Use
RTVMBRD	File	Use
	Library for file	Read
RUNQRY	Query definition, input file, and library	Operational
SAVSAVFDTA	Device description and device file used for save and restore media	Operational
	Save file	Operational
	Library for save file	Read
SAVS36F	From-file	Use
	To-file	All
	Device file or device description	Use
SAVS36LIBM	Save to a physical file	Operational and management
	Either QSYSDKT for diskette or QSYSTAP for tape and all commands need authority to the device	Operational
	Save to a physical file if MBROPT(*ADD) is specified	Add
	Save to a physical file if MBROPT(*REPLACE) is specified	Add and delete
	From-library	Use
STRAPF	Commands CRTPF, CRTLF, ADDPFM, ADDLFM, and RMVM	Use
STRCMTCTL	Message queue	Add
	Data area	Update
	Libraries containing file, message queue, and data area	Read
STRDFU [3]	Create, change, or delete PGM option	Existence
	Display or change data	Operational
	Display data only	Read
	Library to create, change, or delete data file utility program	Read and add
	Change data file	Add, update, and delete
STRIDXSCH	Search index	Use
	Library	Read
UPDDTA	File	Change
	Library	Read
WRKDDMF [3]	DDM file	Operational, management, and existence
	Library	Read and add

Files

Command	Referenced Object	Authority Needed
WRKF [3,5]	Library	Read
WRKQRY [3]		
WRKSCHIDX	Search index	Operational
	Library	Read

Notes:

1 This information refers to physical files only. For more information on these commands, see the *CL Programmer's Guide*.

2 Ownership or operational authority to the file is required.

3 To use individual operations, you must have the authority required by the individual operation.

4 If a new file is created and an authority holder exists for the file, then the user must have all (*ALL) authority to the authority holder or be the owner of the authority holder. If there is no authority holder, the owner of the file is the user who entered the RSTS36F command and the public authority is *ALL.

5 Ownership or some authority for the object is required.

Finance

Command	Referenced Object	Authority Needed
SBMFNCJOB [1]	Job description and message queue	Operational
	Library containing job description and message queue	Read
WRKDEVTBL [1]	Device description	At least one data authority
WRKPGMTBL [1]	Library containing program	Read
WRKUSRTBL [1]		

Note:

1 This command is shipped with authorization provided to only the security officer. Other users must be specifically granted authority to use it.

Forms Control Table

Command	Referenced Object	Authority Needed
ADDFCTE	Forms control table	Operational
	Library for forms control table	Read
CHGFCT	Forms control table	Operational
	Library for forms control table	Read
CHGFCTE	Forms control table	Operational
	Library for forms control table	Read
CRTFCT	Library for forms control table	Add
DLTFCT	Forms control table	Existence
	Library for forms control table	Read
RMVFCTE	Library for forms control table	Use
WRKFCT	Library for forms control table	Use

Graphics Symbol Set

Command	Referenced Object	Authority Needed
CRTGSS	Source file	Use
	Library for graphics symbol set	Read and add
DLTGSS	Graphics symbol set	Existence
	Library for graphics symbol set	Read
WRKGSS [1]	Graphics symbol set	Operational
	Library	Read

Note:

1 Ownership or some authority to the object is required.

Interactive Data Definitions

Command	Referenced Object	Authority Needed
ADDDTADFN	Data dictionary	Change
	File	Operational and management
	Library containing dictionary and file	Read
CRTDTADCT	Library	Read and add
DLTDTADCT	Data dictionary	Operational existence
	Library	Read
DSPDTADCT	Data dictionary	Use
	Library	Read
LNKDTADCT [1]	Data dictionary	Use
	File	Operational and management
	Library	Read
WRKDTADCT [2]	Data dictionary	Operational
	Library	Read
WRKDBFIDD [2]	Data dictionary	Use
	Database file	Operational
	Library	Read
WRKDTADFN [1]	Data dictionary	Use and change
	Library	Read

Notes:

1 Authority to the data dictionary is not required to unlink a file.

2 To use individual operations, you must have the authority required by the individual operation.

Job Descriptions

Command	Referenced Object	Authority Needed
CHGJOBD	Job description	Operational and management
	User profile specified in USER parameter	Operational
	Library	Read
CRTJOBD	User profile specified in USER parameter	Operational
	Library	Read and add

Job Descriptions

Command	Referenced Object	Authority Needed
DLTJOBD	Job description	Existence
	Library	Read
DSPJOBD	Job description	Operational
	Library	Read
WRKJOBD [1]	Job description	Operational
	Library	Read

Notes:

1 Ownership or some authority to the object is required.

Job Queues

Command	Referenced Object	Authority Needed
CLRJOBQ [1,2,3]	Job queue	Read, add, and delete
	Library	Read
CRTJOBQ	Library	Read and add
DLTJOBQ	Job queue	Existence and read
	Library	Read
HLDJOBQ [1,2,3]	Job queue	Read, add, and delete
	Library	Read
RLSJOBQ [1,2,3]	Job queue	Read, add, and delete
	Library	Read
WRKJOBQ [4,5,6]	Job queue	Read
	Library	Read

Notes:

1 If you have job control (*JOBCTL) special authority and OPRCTL(*YES) is specified for the queue, you do not need the authority specified by the AUTCHK parameter.

2 Read, add, and delete authorities are required if the job queue was created with AUTCHK(*DTAAUT) specified. If the queue was created with AUTCHK(*OWNER) specified, you must be the owner of the queue.

3 If you have spool control (*SPLCTL) special authority, you do not need the authority specified by the AUTCHK parameter.

4 If you have job control (*JOBCTL) special authority and OPRCTL(*YES) is specified for the queue, you do not need read authority.

5 If you have spool control (*SPLCTL) special authority, you do not need read authority.

6 When a list of all job queues is requested, only read authority to the library containing the queue is required for the job queue to appear in the list.

Jobs

Command	Referenced Object	Authority Needed
CHGACGCDE [1]		
CHGJOB [1,2,3]	New job queue or output queue if changing job queue or output queue	Read
CHGPJ	User profile for the program start request to specify *PGMSTRRQS	Use
	User profile and job description	Use
	Library	Read

Jobs

Command	Referenced Object	Authority Needed
CHGGRPA [4]	Message queue if associating a message queue with a group	Operational
	Library containing message queue if associating a message queue with a group	Read
DLYJOB [4]		
DSPJOB [1]		
DSPJOBLOG [1,5]		
DSPACTPJ	Subsystem description	Use
	Library	Read
ENDJOBABN [1]		
ENDPJ [6]	Library	Read
HLDJOB [1]		
RLSJOB [1]		
RRTJOB	Subsystem description	Operational
RTVJOBA [1,7]		
SBMDBJOB	Database file	Use
	Job queue	Read
	Library containing database file and job queue	Read
SBMDKTJOB	Message queue	Use
	Job queue and device description	Operational and add
	Library containing message queue and job queue	Read
SBMJOB [2]	Subsystem description and message queue	Operational
	Job description	Use and add
	Message queue	Use and add
	Libraries specified in the initial library list in the job description	Read
	User profile specified in user keyword	Use
SBMNETJOB	Job	Use
	Library	Read
STRPJ [6]	Library	Read
TFRGRPJOB	Initial group program	Operational
	Library containing initial group program	Read
TFRJOB [8]	Job queue and for subsystem that the job queue is allocated to	Read and add

Jobs

Command	Referenced Object	Authority Needed
WRKJOB [9]		

Notes:

1 Any user can run these commands for jobs running under his own user profile. A user with job control (*JOBCTL) special authority can run these commands for any job.

2 You must have the authority (specified in your user profile) for the scheduling priority and output priority specified.

3 To change certain job attributes even in the user's own job requires job control (*JOBCTL) special authority. These attributes are EXCPTY, TIMESLICE, PURGE, and DFTWAIT.

4 This command only affects the job in which it was specified.

5 To display the log for a job having security officer authority, you must also have security officer (*ALLOBJ) special authority.

6 To use this command, job control *JOBCTL special authority is required.

7 Authority is verified at run time but not at compile time.

8 If the job being transferred is an interactive job, the following restrictions apply:

- The job queue where the job is placed must be associated with an active subsystem.
- The work station associated with the job must have a corresponding work station entry in the subsystem description associated with the new subsystem.
- The work station associated with the job must not have another job associated with it that has been suspended by means of the Sys Req (System Request) key. The suspended job must be canceled before the Transfer Job command can run.
- The job must not be a group job.

9 The user must either own the object or must have operational and read authorities for the specified libraries and any one of the object authorities for each of the objects. If the user does not have read authority for a library, none of the objects are shown.

Journals

Command	Referenced Object	Authority Needed
APYJRNCHG	Journal	Use
	Library for journal	Read
	Journal receiver	Use
	Library for journal receiver	Read
	Files whose journaled changes are being applied or removed	Change and management
	Library for files	Read
CHGJRN	Journal receiver if specified	Use
	Library for new receiver	Read
	Attached journal receiver	Use and management
	Library for attached journal receiver	Read
	Journal	Operational, management, and update
	Library for journal	Read
	Library containing message queue	Read
CMPJRNIMG	Journal	Use
	Library for journal	Read
	Journal receiver	Use
	Library for journal receiver	Read
	File	Use
	Library for file	Read

Journals

Command	Referenced Object	Authority Needed
CRTJRN	Library for journal	Read and add
	Journal receiver	Use and management
	Library for journal receiver	Read
DLTJRN	Journal	Operational and existence
	Library for journal	Read
DSPJRN	Journal	Use
	Journal if FILE(*ALLFILE) is specified or the specified file has been deleted from the system	Use and existence
	Library for journal	Read
	Journal receiver	Use
	File if specified	Use
	Library for file	Read
	Outfile if it exists	Operational, management, add, and delete
	Library for outfile	Read and add
DSPJRNMNU[2]		
ENDJRNAP	Journal	Management
	Library for journal	Read
	File	Operational and management
	Library for file	Read
ENDJRNPF	Journal	Management
	Library for journal	Read
	File	Use and management
	Library for file	Read
RCVJRNE	Journal	Use
	Journal if FILE(*ALLFILE) is specified or the specified file has been deleted from the system	Use and existence
	Library for journal	Read
	Journal receiver	Use
	File	Use
	Library for file	Read
	Exit program	Read
RMVJRNCHG	Journal	Use
	Library for journal	Read
	Journal receiver	Use
	Library for journal receiver	Read
	Files whose journaled changes are being applied or removed	Change and management
	Library for files	Read

Journals

Command	Referenced Object	Authority Needed
RTVJRNE	Journal	Use
	Journal if FILE(*ALLFILE) is specified or the specified file has been deleted from the system	Use and existence
	Library for journal	Read
	Journal receiver	Use
	File	Use
	Library for file	Read
SNDJRNE	Journal	Operational and add
	Library for journal	Read
	File if specified	Operational
	Library for file	Read
STRJRNAP	Journal	Operational and management
	Library for journal	Read
	File	Use and management
	Library for file	Read
STRJRNPF	Journal	Operational and management
	Library for journal	Read
	File	Operational and management
	Library for file	Read
WRKJRN [1]	Journal	Use
	Library for journal	Read
	Journal receiver if receiver information is requested	Use
	File if file processing is requested	Read
	Library for file if forward or backout recovery is requested	Use
	Objects that are deleted during recovery	Existence
WRKJRNA	Journal	Use
	Library for journal	Read
	Journal receiver	Operational and read
	Library for journal receiver	Read

Note:

1 Additional authority is required for specific functions called during the operation selected. For example, to restore an object, the user needs special authority. The user also needs the authority required for any command called during a specific function.

2 See the WRKJRN command (this command has the same function)

Journal Receivers

Command	Referenced Object	Authority Needed
CRTJRNRCV	Library for journal receiver	Read and add

Journal Receivers

Command	Referenced Object	Authority Needed
DLTJRNRCV	Journal receiver	Operational and existence
	Library for journal receiver	Read
	Journal	Operational
	Library for journal	Read
DSPJRNRCVA	Journal receiver	Operational and some data authority
	Library for journal receiver	Read
WRKJRNRCV	Journal receiver	Operational
	Journal receiver (Delete option)	(see DLTJRNRCV)
	Journal receiver (Display Description option)	(see DSPJRNRCVA)
	Journal receiver (Change Description option)	Management and read
	Library for journal receiver	Read

Languages

Command	Referenced Object	Authority Needed
CRTBASPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTCLPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTCBLPGM (COBOL/400* licensed program)	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTCBLPGM (S/38 environment)	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTCPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTFTNPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTPASPGM	Library	Read and add
CRTPLIPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add

Languages

Command	Referenced Object	Authority Needed
CRTRMCPGM	Source file	Use
	Externally described device and database files referenced in program	Operational
	Library	Read and add
CRTRPGPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTS36CBL (S/36 environment)	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTS36RPG	Source file	Use
	Library	Read and add
CRTS36RPGR	Source file	Use
	Library	Read and add
CRTSQLC (SQL/400* licensed program) [1,2] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLCBL (SQL/400 licensed program) [1,2] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLFTN [1, 2] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLPLI (SQL/400 licensed program) [1,2] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLRPG (SQL/400 licensed program) [1,2] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
STRBAS	Externally described device and database files referenced in source program	Operational
	Source file	Read, add, update, and delete
	Library	Read and add
STRBASPRC		
STRCBLDBG	Program	Change
STRREXPRC	Source file	Read
	Library	Read
STRSQL (SQL/400 licensed program) [1]	Data description specifications	Operational
	Library	Read and add

Notes:

1 The *SQL/400* Reference* contains more information about security requirements for structured query language (SQL) statements.

2 If a value is specified for RDBNAME parameter, use authority is needed to the CRTSQLPKG command.

Libraries

Command	Referenced Object	Authority Needed
ADDLIBLE	Library	Use
CHGCURLIB	Current library	Use
CHGLIB	Library	Management
CHGLIBL	Every library being placed in the library list	Use
CHGSYSLIBL [1]	Library	Use
CLRLIB [2]	Library	Read
	Every object being deleted from library	Existence
CPYLIB[4]	Library	Use
	To-library	Use and add
	Objects being copied	Use
DLTLIB [2]	Library	Use and existence
	Every object in the library	Existence
DSPLIB	Library	Read
DSPLIBD	Library	Some authority other than *EXCLUDE
EDTLIBL	Library to add to list	Use
RSTLIB [3]	Library	Read and add
	Message queues being restored to library where they already exist	Operational
	Library saved if VOL(*SAVVOL) is specified.	Use
	Every object in library	Existence
	User profile owning objects being created	Add
RSTS36LIBM	From-file	Use
	To-file	Change
	To-library	Change
	Device file or device description	Use
SAVLIB [3]	To-library	Change
	Every object in the library	Existence
SAVS36LIBM	Save to a physical file	Operational and management
	Either QSYSDKT for diskette or QSYSTAP for tape, and all commands need authority to the device	Operational
	Save to a physical file if MBROPT(*ADD) is specified	Add
	Save to a physical file if MBROPT(*REPLACE) is specified	Add and delete
	From-library	Use
RTVLIBD	Library	Some authority other than *EXCLUDE

Libraries

Command	Referenced Object	Authority Needed
WRKLIB	Library	Read
	Library QSYS	Read

Notes:

1 This command is shipped with authorization provided to only the security officer. Other users must be specifically given authority to use this command.

2 If object existence is not found for some objects in the library, those objects are not deleted, and the library is not completely cleared and deleted. Only authorized objects are deleted.

3 If you have save system (*SAVSYS) special authority, you can save and restore any object. When saving to tape or diskette, you must have object operational authority to the device description and the device file for the tape or diskette. When saving to a save file, you must have object operational and add authorities to the save file. When restoring from a save file, you must have object operational and read authorities to the save file.

4 All restrictions that apply to the CRTDUPOBJ command, also apply to this command.

Licensed Program

Command	Referenced Object	Authority Needed
DLTLICPGM [1,2,3]		
RSTLICPGM [1,2,3]		
SAVLICPGM [1,2,3]		

Notes:

1 Some licensed programs can be deleted, saved, or restored only if you are enrolled in the system distribution directory.

2 You must have the security officer grant you authority to use this command.

3 If deleting, restoring, or saving a licensed program that contains folders, all restrictions that apply to the DLTDLO command also apply to this command.

Line Descriptions

Command	Referenced Object	Authority Needed
CHGLINASC	Line description	Change
	The controller description specified in the SWTCTLLST parameter	Use
CHGLINBSC	Line description	Change
	The controller description specified in the SWTCTLLST parameter	Use
CHGLINIDLC	The connection list in the CNNLSTIN parameter	Use
	The network interface description specified in the SWTNWILST parameter	Use
	The control description in the CTL parameter	Use
CHGLINETH	Line description	Change
CHGLINSDLC	Line description	Change
CHGLINTDLC	Line description	Change
CHGLINTRN	Line description	Change
CHGLINX25	Line description	Change
	The controller description specified in the SWTCTLLST parameter	Use

Line Descriptions

Command	Referenced Object	Authority Needed
CRTLINASC	The controller description specified in the CTL and SWTCTLLST parameters	Use
CRTLINBSC	The controller description specified in the SWTCTLLST parameter and the CTL parameter	Use
CRTLINETH	The controller description specified in the NETCTL parameter	Use
CRTLINIDLC	The connection list in the CNNLSTIN parameter	Use
	The network interface description in NWI or SWTNWILST parameter	Use
	The control description in the CTL parameter	Use
CRTLINSDLC	The controller description specified in the SWTCTLLST parameter and the CTL parameter	Use
CRTLINTDLC	The controller description specified in the WSC parameter and the CTL parameter	Use
CRTLINTRN	Controller description in the NETCTL parameter	Use
CRTLINX25	The controller description specified in the SWTCTLLST parameter	Use
	Permanent virtual circuit (PVC) controller description specified in the LGLCHLE parameter	Use
DLTLIND	Line description	Operational and existence
DLTSUPQS	QAQABBPY	Read
DSPLIND	Line description	Use
ENDLINRCY	Line description	Operational
RSMLINRCY	Line description	Operational
VFYCMN [1]	Object	Use
WRKLIND [2]	Line description	Operational

Notes:

1 Only the QSYSOPR, QPGMR, QSRV, QSRVBAS, or QSECOFR user profiles can use this command.

2 To use individual operations, you must have the authority required by the individual operation.

Media

Command	Referenced Object	Authority Needed
CRTTBL	Library	Read and add
DLTTBL	Table	Existence
	Library	Read
INZDKT	Diskette	Management

Menu and Panel Group

Command	Referenced Object	Authority Needed
CHGMNU	Menu	Change
	Library	Use
CRTMNU	Library for menu	Use
CRTS36MNU	Library for menu	Read and add
	Source file	Use
	Library for source file	Read
	Message files named in source	Operational and existence
	To-file source file when TOMBR is not *NONE	Operational, management, existence, and add
	Library for to-file	Read and add
	Menu display file when REPLACE(*YES) is specified	Operational and existence
	Command text message file	Operational and existence
	Create Message File (CRTMSGF) command	Operational
	Add Message Description (ADDMSGD) command	Operational
	Create Display File (CRTDSPF) command	Operational
DLTMNU	Menu	Existence
	Library	Use
DLTPNLGRP	Panel group	Existence
	Library for panel group	Read
DSPMNUA	Menu	Use
	Library	Read
GO	Menu	Use
	Library for menu	Read
	Display file and message files with *DSPF specified	Use
	Display file and program with *PGM specified	Use
WRKMNU	Menu	Operational
	Library for menu	Operational
WRKPNLGRP	Panel group	Operational
	Library for panel group	Read

Message Description

Command	Referenced Object	Authority Needed
ADDMSGD	Message file	Use and add
	Library	Read
CHGMSGD		
DSPMSGD	Message file	Use and update
	Library	Read
RMVMSGD	Message file	Use and delete
	Library	Read

Message Description

Command	Referenced Object	Authority Needed
WRKMSGD	Message file	Use
	Library	Read
	Add message to message file	Add
	Change message in message file	Update
	Remove message from message file	Delete

Message Files

Command	Referenced Object	Authority Needed
CRTMSGF	Library	Read and add
DLTMSGF	Message file	Existence
	Library	Read
DSPMSGF	Message file	Use
	Library	Read
MRGMSGF	Message files	Operational
	Library	Read
	From-message file	Use
	Library	Read
	To-message file	Use, add, and delete
	Library	Read
	Replace-message file	Use, add
	Library	Read
WRKMSGF	Message file	Operational
	Library for message file	Read

Message Queues

Command	Referenced Object	Authority Needed
CHGMSGQ	Message queue	Use and delete
	Library	Read
CLRMSGQ	Message queue	Operational and delete
	Library	Read
CRTMSGQ	Library	Read and add
DLTMSGQ	Message queue	Operational, existence and delete
	Library	Read
DSPLOG	History log (QHST)	None
WRKMSGQ	Message queue	Operational
	Library	Use

Messages

Command	Referenced Object	Authority Needed
DSPMSG	Message queue	Use
	Library	Use
	Reply to inquiry messages	Use and add
	Remove messages from message queue	Use and delete
RCVMSG	Message queue	Use
	Library	Read
RMVMSG	Message queue	Operational and delete
	Library	Read
RTVMSG	Message file	Use
	Library	Read
SNDBRKMSG	Message queue	Operational and add
SNDMSG	Message queue	Operational and add
	Library	Read
SNDPGMMSG	Message queue	Operational and add
	Library	Read
SNDRPY	Message queue	Use and add
	Library	Read
	Remove messages from queue	Use, add, and delete
SNDUSRMSG	Message queue	Operational and add
	Library	Read

Mode Description

Command	Referenced Object	Authority Needed
CHGMODD	Mode description	Change
CHGSSNMAX	Device description	Operational
DLTMODD	Mode description	Operational and existence
DSPMODD	Mode description	Read
DSPMODSTS	Mode description	Operational
ENDMOD	Device description	Operational
STRMOD	Device description	Operational
WRKMODD [1]	Mode description	Operational

Note:

1 To use individual operations, you must have the authority required by the individual operation.

Network Attributes

Command	Referenced Object	Authority Needed
ADDNETJOBE		
CHGNETA [1]		
CHGNETJOBE		
DLTNETF		

Network Attributes

Command	Referenced Object	Authority Needed
DSPAPPNINF		Operational
	Library	Add and read
RCVNETF		Operational
	Library	Add and read
RMVNETJOBE		
RMVSOCE	Sphere of control	Operational and delete
SBMNETJOB	Job	Use
	Library	Read
SNDNETF		Use
	Library	Read
SNDNETSPLF	Spool file	Read
	Library	Read
WRKALR	Alert table	Use and delete
	Library	Read
WRKNETF [2]		
WRKNETJOBE [2]	Job entry	Read
	Library	Read
WRKSOC	Sphere of control	Use
WRKSOCE	Sphere of control	Use and add

Notes:

1 The sphere of control is physical file QSYS/QAALSOC.

2 Additional authorities are required for specific functions called by the operations selected. The user also needs additional authorities for any commands called during a specific function.

Network Interface Descriptions

Command	Referenced Object	Authority Needed
CHGNWIISDN	Network interface description	Change
	The line description specified in the CHLENTRY parameter	Use
CRTNWIISDN	Network interface description	Use
	The line description specified in the CHLENTRY parameter	Use
DLTNWID	Network interface description	Operational and existence
DSPNWID	Network interface description	Use
WRKNWID [1]	Network interface description	Use

Note:

1 To use the individual operations, you must have the authority required by the individual operation.

Office

Command	Referenced Object	Authority Needed
ADDACC [1]		Management, read, add, update, and delete
GRTACCAUT [1,2,3]		

Office

Command	Referenced Object	Authority Needed
GRTUSRPMN [1,2]	In-folder	Change
RMVACC [1]		
RVKACCAUT [1]		
RVKUSRPMN [1,2]	In-folder	Operational
	In-folder	Change
WRKDOCPRTQ [4]		
WRKFLR	Folder	Use
WRKTXTPRF	Document SYSTEM in folder	Use

Notes:

1 You must have all object (*ALLOBJ) or security administrator (*SECADM) special authority to grant or revoke access code authority or document authority for other users.

2 Access is restricted to documents, folders, and mail that is not personal.

3 The access code must be defined to the system (using the Add Access Code (ADDACC) command) before you can grant access code authority. The user being granted access code authority must be enrolled in the system distribution directory.

4 Additional authorities are required for specific functions called by the operations selected. The user also needs additional authorities for any commands called during a specific function.

Online Education

Command	Referenced Object	Authority Needed
CVTEDU [1]		
STREDU [2]		

Notes:

1 Only the security officer user profile (QSECOFR) or a user with security administrator special authority can use this command.

2 Only the security officer user profile (QSECOFR) or a user with security administrator special authority can display options for the Education Administrator menu.

Operational Assistant

Command	Referenced Object	Authority Needed
CHGCLNUP[1]		
ENDCLNUP[2]		
STRCLNUP[2]		

Notes:

1 You must have all object (*ALLOBJ) and job control (*JOBCTL) special authorities specified in your user profile.

2 You must have job control (*JOBCTL) special authority specified in your user profile.

OSI Communications Subsystem/400

Command	Referenced Object	Authority Needed
ADDOSIABSN[1]	Metatable file	Use
ADDOSIADJN[1]		
ADDOSIAGT[1]		

OSI Communications Subsystem/400

Command	Referenced Object	Authority Needed
ADDOSIAGTR[1]		
ADDOSIAPPE[1]		
ADDOSIAPPM[1]		
ADDOSIAPPX[1]		
ADDOSIAUNN[1]		
ADDOSICLPS[1]		
ADDOSICMPS[1]		
ADDOSIDUAR[1]		
ADDOSIIX25[1]		
ADDOSILINE[1]	Line description	Use
ADDOSILINS[1]		
ADDOSIMGR[1]		
ADDOSIMGRR[1]		
ADDOSINSAP[1]		
ADDOSIOX25[1]		
ADDOSIQOSM[1]		
ADDOSIRTE[1]		
ADDOSISSEL[1]		
ADDOSISUBN[1]		
ADDOSITPTM[1]		
CHGOSIABSN[1]	Metatable file	Use
CHGOSIADJN[1]		
CHGOSIAPPE[1]		
CHGOSIAPPM[1]		
CHGOSIAPPX[1]		
CHGOSIAUNN[1]		
CHGOSICLPS[1]		
CHGOSICMPS[1]		
CHGOSIDUAR[1]		
CHGOSIIX25[1]		
CHGOSILCLA[1]		
CHGOSILINE[1]	Line description	Use
CHGOSILINS[1]		
CHGOSIMGRR[1]		
CHGOSINSAP[1]		
CHGOSIOX25[1]		
CHGOSIQOSM[1]		
CHGOSIRTE[1]		
CHGOSISSEL[1]		
CHGOSISUBN[1]		
CHGOSITPTM[1]		
CRTLASREP[6]	Input file	Use
	Metatable file	Change and management
	Data structures file	Change and management

OSI Communications Subsystem/400

Command	Referenced Object	Authority Needed
DSPOSISAP[3]		
ENDOSI[4]		
ENDOSIASN [5]		
ENDOSINL[5]		
RMVOSISBSN[1]		
RNVOSIADJN[1]		
RMVOSIAGT[2]		
RMVOSIAGTR[1]		
RMVOSIAPPE[1]		
RMVOSIAPPM[1]		
RMVOSIAPPX[1]		
RMVOSIAUNN[1]		
RMVOSICLPS[1]		
RMVOSICMPS[1]		
RMVOSIDUAR[1]		
RMVOSIOX25[1]		
RMVOSILINE[1]		
RMVOSILINS[1]		
RMVOSIMGR[2]		
RMVOSIMGRR[1]		
RMVOSINSAP[1]		
RMVOSIOX25[1]		
RMVOSIQOSM[1]		
RMVOSIRTE[1]		
RMVOSISSEL[1]		
RMVOSISUBN[1]		
RMVOSITPTM[1]		
SETOSIATR[2]		
STROSINL[5]		
TRCOSIASN[3]		
TRCOSIPCL[3]		

Notes:

1 The public is not authorized to this command. IBM-supplied user profiles that have authority to use this command are QPGMR, QSRV, QSRVBAS, QSYSOPR, and QSECOFR.

2 The public is not authorized to this command. IBM-supplied user profiles that have authority to use this command are QSRV, QSRVBAS, QSYSOPR, and QSECOFR.

3 The public is not authorized to this command. IBM-supplied user profiles that have authority to use this command are QSRV, QSRVBAS, and QSECOFR.

4 The public is not authorized to this command. IBM-supplied user profile that has authority to use this command is QSECOFR.

5 The public is not authorized to this command. IBM-supplied user profiles that have authority to use this command are QSYSOPR, and QSECOFR.

6 The public is not authorized to this command. IBM-supplied user profiles that have authority to use this command are QPGMR and QSECOFR.

Output Queue

Command	Referenced Object	Authority Needed
CHGOUTQ [1,2]	Output queue	Management, read, add, and delete
	Library for output queue	Read
CLROUTQ [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
CRTOUTQ	Library for output queue	Read and add
DLTOUTQ	Output queue	Existence and read
	Library for output queue	Read
HLDOUTQ [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
RLSOUTQ [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
WRKOUTQ [4,5,6]	Output queue	Read
	Library for output queue	Read
WRKOUTQD [4,5]	Output queue	Read
	Library for output queue	Read

Notes:

1 If you have spool control (*SPLCTL) special authority, you do not need the authority specified by the AUTCHK parameter.

2 Read, add, delete authorities are required if the queue was created with AUTCHK(*DTAAUT) specified. You must be the owner of the queue if AUTCHK(*OWNER) is specified.

3 If you have job control (*JOBCTL) special authority and OPRCTL(*YES) is specified for the queue, you do not need the authority specified by the AUTCHK parameter.

4 If you have job control (*JOBCTL) special authority and OPRCTL(*YES) is specified for the queue, you do not need read authority.

5 If you have spool control (* SPLCTL) special authority, you do not need read authority.

6 When a list of all output queues is requested, only read authority to the library containing the queue is required for the output queue to appear in the list.

Packages

Command	Referenced Object	Authority Needed
CRTSQLPKG (For Version 2 Release 1.1)	Program	Use
	Library	Read
DLTSQLPKG (For Version 2 Release 1.1)	Package	Existence and management
	Library	Read

Performance

Command	Referenced Object	Authority Needed
ANZACCGRP [1]		
ANZDBF [1]		
ANZDBFKEY [1]		
ANZPGM [1]		
DSPACCGRP [1]		
DSPPFRDTA [1]		
ENDJOBTRC [1]		

Performance

Command	Referenced Object	Authority Needed
ENDSAM [1]		
ENDSAMCOL [1]		
MDLSYS [1]		
PRTPOLRPT [1]		
PRTACTRPT [1]		
PRTCPTRPT [1]		
PRTJOBRPT [1]		
PRTJOBTRC [1]		
PRTLCKRPT [1]		
PRTRSCRPT [1]		
PRTSAMDTA [1]		
PRTSYSRPT [1]		
PRTTNSRPT [1]		
STRJOBTRC [1]		
STRPFRT [1]		
STRSAM [1]		
STRSAMCOL [1]		
WRKSYSACT [1]		

Note:

1 Only QPGMR, QSRV, QSRVBAS, or QSECOFR user profiles can use this command.

Problem

Command	Referenced Object	Authority Needed
ANZPRB [1]		
CHGPRB [1]		
DLTPRB [1]		
VFYCMN [1]		
VFYTAP [1]		
VFYPRT [1]	Device description	Use
WRKPRB [1]	Line, controller, and device based on problem analysis action	Use and add

Note:

1 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QPGMR, QSRV, QSRVBAS, QSYSOPR, and QSECOFR.

Programs

Command	Referenced Object	Authority Needed
ADDPGM [1]	Program	Change
	Library	Add and read
CALL [1]	Program	Operational and one data authority
	Some functions when using high-level languages	Read if these functions are used

Programs

Command	Referenced Object	Authority Needed
CHGDBG	Debug operation	Use, add and delete
	Program	Change
CHGPGM	Program	Management
	Library	Read
	Library if re-create operation is requested	Read and delete
CRTBASPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTCPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTCBLPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTFTNPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTPASPGM	Library	Read and add
CRTPLIPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTRMCPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTRPGPGM	Source file	Use
	Externally described device and database files referenced in source program	Operational
	Library	Read and add
CRTRPTPGM	Source file	Use
	Library	Read and add
CRTS36RPT	Source file	Use
	Library	Read and add
CRTS36RPG	Source file	Use
	Library	Read and add
CRTS36RPGR	Source file	Use
	Library	Read and add
CRTSQLC (SQL/400 licensed program) [2,3] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add

Programs

Command	Referenced Object	Authority Needed
CRTSQLCBL (SQL/400 licensed program) [2,3] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLFTN [3] (For Version 2 Release 1.1)	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLPLI (SQL/400 licensed program) [3]	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CRTSQLRPG (SQL/400 licensed program) [2,3]	Source file	Use
	Data description specifications	Operational
	Library	Read and add
CVTCLSRC	From-file	Read
	To-file	Add
DLTDFUPGM	Program	Existence
	Library	Read
DLTPGM	Program	Existence and management
	Library	Read
DMPCLPGM	CL program	Read
	Library	Read
DSPPGM	Program	Use
	Library	Read
DSPPGMREF	Program	Operational
	File specified on the OUTFILE parameter	Operational, management, add, and delete
	CRTPF command if CRTFILE(*YES) is specified	Operational
	Library specified on the OUTFILE parameter if file does not exist	Add
	Library specified on the OUTFILE parameter if file does exist	Read
ENDCBLDBG		Change
EXTPGMINF	Source file and database files	Operational
	Library	Read and add
RTVCLSRC		Use and management
	Database source file	Operational, management, add, and delete
	Library for source file	Read
SETATNPGM	Attention-key-handling-program or one or more data authorities	Operational or one or more data authorities
	Library containing program	Read
SETPGMINF	Database files	Operational
	Source file	Use
	Library	Read and add
	Program for which the control information is being set	Read and update
STRCBLDBG	Program	Change

Programs

Command	Referenced Object	Authority Needed
STRDBG	Program	Change
STRSQL (SQL/400 licensed program) [2]	Data description specifications	Operational
	Library	Read and add
TFRCTL	Program or one of the data authorities	Use or one data authority
	Some language functions when using high-level languages	Read if these functions are used
WRKPGM	Program	Operational
	Library for program	Read

Notes:

1 When a program is in a debug operation, no further authority is needed for debug commands.

2 The *SQL/400* Reference* contains more information about security requirements for SQL statements.

3 If a value is specified for the RDBNAME parameter, use authority is needed to the CRTSQLPKG command.

Query Manager Forms and Queries

Command	Referenced Object	Authority Needed
WRKQMFORM [1]	Query manager form	Operational
	Library	Read
WRKQMQRY [1]	Query manager query	Operational
	Library	Read

Note:

1 Ownership or some authority to the object is required.

Question and Answer

Command	Referenced Object	Authority Needed
ANSQST	Database file QAQAXXBQPY	Read
ASKQST	Database file QAQAXXBBPY	Read
CHGQSTDB	Database file QAQAXXBQPY	Read
CRTQSTDB	Database file QAQAXXBQPY	Read
CRTQSTLOD	Database file QAQAXXBQPY	Read
DLTQST	Database file QAQAXXBQPY	Read
DLTQSTDB	Database file QAQAXXBQPY	Read
EDTQST	Database file QAQAXXBQPY	Read
LODQSTDB	Database file QAQAXXBQPY	Read
STRQST	Database file QAQAXXBBPY	Read
WRKQST	Database file QAQAXXBBPY	Read

Reader

Command	Referenced Object	Authority Needed
ENDRDR [1]		
HLDRDR [1]		

Reader

Command	Referenced Object	Authority Needed
RLSRDR [1]		
STRDBRDR	Message queue	Operational and add
	Database file and job queue	Read
	Library containing job queue, database file, and message queue	Read
STRDKTRDR	Message queue	Operational and add
	Library containing job queue and message queue	Read
	Job queue and device description	Read

Note:

1 You must be the user who started the reader, or you must have all object (*ALLOBJ) or job control (*JOBCTL) special authority.

Relational Database Directory

Command	Referenced Object	Authority Needed
DPRDBDIRE	File specified on OUTFILE parameter	Operational, management, add, and delete
	Library specified on OUTFILE parameter if file does not exist	Add
	Library specified on OUTFILE parameter if file does exist	Read
	CRTPF command if OUTFILE parameter specified and it does not exist	Operational
WRKRDBDIRE	File specified on OUTFILE parameter	Operational, management, add, and delete
	Library specified on OUTFILE parameter if file does not exist	Add
	Library specified on OUTFILE parameter if file does exist	Read
	CRTPF command if OUTFILE parameter specified and it does not exist	Operational

Service

Command	Referenced Object	Authority Needed
APYPTF [3]	Product library	Object management
CHGSRVPVDA [7]	Data area QNSATTR	Change

Service

Command	Referenced Object	Authority Needed
CPYPTF [2,3]	To file	Change
	From file	Use
	From library	Use
	Tape device	Use
	Diskette device	Use
	To library	Use
	QSRV	Use
	CRTLIB command	Use
	LICPGM library	Use
	OVRTAPF command	Use
	CPYTOTAP command	Use
	CHKTAP command	Use
	CPYFRMTAP command	Use
	CRTSAVF command	Use
	CRTTAPF command	Use
	CRTPF command	Use
CRTAPAR [1,2,5]		
DMPJOB [5]		
DMPJOBINT [5]		
DSPPTF [3]	QGPL	Use
	Product library	Use
DSPSRVSTS [3]		
ENDCPYSCN	Device description	Use
ENDSRVJOB [5]		
LODPTF [2,3]	Object	Operational and management
	QGPL Library	Management
	QSYS Library	Management
	Licensed program library	Management
	Tape or diskette	Use
PRTERRLOG [3]		
PRTINTDTA [3]		
RMVPTF [3]	Object	Operational and management
	Product library	Management
SNDPTFORD [7]		
SNDSRVRQS [7]		
STRCPYSCN	Job queue	Use
	Device description	Use
STRSRVJOB [3]	User profile of job	Use
STRSST [4]		
TRCINT [6]		
TRCJOB [5]		
VFYCMN [3]		
VFYPRT [3]	Device description	Use
VFYTAP [3]	Device description	Use

Service

Command	Referenced Object	Authority Needed
WRKCNTINF [7]		
WRKPRB [1,3]	Line, controller, and device based on problem analysis action	Use and add
WRKSRVPVD [7]	Database file QANSSRI	Change
WRKSRVRQS [7]	Database file QAEDSPI	Change

Notes:

1 You need authority to the PRTERRLOG command for some analysis procedures or if the error log records are being saved.

2 All restrictions for the RSTOBJ command also apply.

3 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QPGMR, QSRV, QSRVBAS, QSYSOPR, and QSECOFR.

4 Service (*SERVICE) special authority is required to run this command.

5 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QPGMR, QSRV, QSRVBAS, QSYSOPR, and QSECOFR.

6 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QPGMR, QSRV, and QSECOFR.

7 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QSECOFR, QSRV, and QSRVBAS.

Session Descriptions

Command	Referenced Object	Authority Needed
ADDRJECMNE	Library	Read
ADDRJERDRE	Library	Read
ADDRJEWRTE	Library	Read
CHGRJECMNE	Library	Read
CHGRJERDRE	Library	Read
CHGRJEWTRE	Library	Read
CHGSSND	Library	Read
CNLRJERDR	Library	Read
CNLRJEWTR	Library	Read
CRTRJEBSCF	Library	Add
CRTRJECFG	Library	Add
CRTRJECMNF	Library	Add
CRTSSND	Library	Add
CVTRJEDTA	Library	Operational
DLTRJECFG	Configuration object	Existence
	Library	Read
DSPRJECFG	Configuration object	Operational
	Library	Read
ENDRJESSN	Library	Read
RMVRJECMNE	Library	Read
	Session description	Operational
RMVRJERDRE	Library	Read
	Session description	Operational

Session Descriptions

Command	Referenced Object	Authority Needed
RMVRJEWTRE	Library	Read
	Session description	Operational
SBMRJEJOB	Library	Read
	Input database file and any files named on embedded READFILE control statements	Use
	Session description	Operational
STRRJECSL	Library	Read
	Session description	Operational
STRRJERDR	Library	Read
	Session description	Operational
STRRJESSN	Library	Read
	Session description	Operational
STRRJEWTR	Library	Read
	Session description	Operational
WRKRJESSN	Library	Read
	Session description	Operational
WRKSSND	Library	Read
	Session description	Operational

Spelling Aid Dictionaries

Command	Referenced Object	Authority Needed
CRTSPADCT	Spelling aid dictionary	Existence
DLTSPADCT	Spelling aid dictionary	Existence
	Library	Read
WRKSPADCT	Spelling aid dictionary	Operational
	Library	Read

Spooled Files

Command	Referenced Object	Authority Needed
CHGSPLFA [1,2,3,5,6,7]	Output queue	Read, add, and delete
	Library for output queue	Read
CPYSPLF [1,2,3,4,5]	Output queue	Read, add, and delete
	Database file if MBROPT(*REPLACE) is specified	Operational, management, add and delete
	Database file if MBROPT(*ADD) is specified	Operational and add
	Library for database file	Read
DLTSPLF [1,2,3,5]	Output queue	Read, add, and delete
DSPSPLF [1,2,3,4,5]	Output queue	Read, add, and delete
HLDSPLF [1,2,3,5]	Output queue	Read, add, and delete
RLSSPLF [1,2,3,5]	Output queue	Read, add, and delete
RCLSPLSTG [8]		

Spooled Files

Command	Referenced Object	Authority Needed
SNDNETSPLF [1,2,3,4,5]	Output queue	Read, add, and delete

Notes:

1 Read, add, and delete authorities are required if the queue was created with AUTCHK(*DTAAUT) specified. If AUTCHK(*OWNER) is specified, you must be the owner of the queue.

2 If you have job control (*JOBCTL) special authority and the operator control authority is OPRCTL(*YES) on the queue, you do not need the authority specified on the AUTCHK parameter.

3 If you have spool control (*SPLCTL) special authority, you do not need the authority specified by the AUTCHK parameter.

4 Only read authority is required if DSPDTA(*YES) is specified on the queue.

5 Users are always authorized to control their own spooled files.

6 To move a spooled file to another output queue, you must have read authority to the new output queue, job control authority (*JOBCTL) and operator control authority specified as OPRCTL(*YES) for the new output queue, or spool control authority (*SPLCTL) and read authority to the new output queue library.

7 To move a spooled file to the front of an output queue (PRTSEQ(*NEXT)) or to change its priority to a value greater than the limit specified in your user profile, you must have either [1], [2], or [3].

8 The public is not authorized to this command. The IBM-supplied user profiles that have authority to this command are QPGMR, QSYSOPR, QSRV, QSRVBAS, and QSECOFR.

Subsystem Descriptions

Command	Referenced Object	Authority Needed
ADDAJE		Operational and management
	Library	Read
ADDPJE	User profile for the program start request to specify *PGMSTRRQS	Use
	User profile and job description	Use
	Library	Read
ADDCMNE		Operational and management
	Library	Read
ADDJOBQE	Job queue	Operational and management
	Library	Read
ADDRTGE		Operational and management
	Library	Read
ADDWSE		Operational and management
	Library	Read
CHGAJE		Operational and management
	Library	Read
CHGCMNE		Operational and management
	Library	Read
CHGJOBQE	Job queue	Operational and management
	Library	Read
CHGPJE	User profile for the program start request to specify *PGMSTRRQS	Use
	User profile and job description	Use
	Library	Read
CHGRTGE		Operational and management
	Library	Read

Subsystem Descriptions

Command	Referenced Object	Authority Needed
CHGSBSD		Operational and management
	Library	Read
CHGWSE		Operational and management
	Library	Read
CRTSBSD	Library	Read and add
DLTSBSD		Operational and existence
	Library	Read
DSPSBSD		Operational
	Library	Read
ENDSBS		Operational
	Library	Read
ENDSYS [1]		
PWRDWNSYS [1]		
RMVAJE		Operational and management
	Library	Read
RMVCMNE		Operational and management
	Library	Read
RMVJOBQE	Job queue	Operational and management
	Library	Read
RMVPJE	Subsystem description	Use
	Library	Read
RMVRTGE		Operational and management
	Library	Read
RMVWSE		Operational and management
	Library	Read
WRKSBSD		Operational and management
	Library	Read

Note:

1 You must have job control (*JOBCTL) special authority to use this command.

System Reply List

Command	Referenced Object	Authority Needed
ADDRPYLE [1]		
CHGRPYLE [1]		
RMVRPYLE [1]		
WRKRPYLE [1]		

Note:

1 This command is shipped authorized to the QPGMR user profile.

System Values

Command	Referenced Object	Authority Needed
CHGSYSVAL [1,2]		

Notes:

1 Certain system values can only be changed by jobs with security officer authority (*ALLOBJ and *SECADM special authorities).

2 This command can only be used by QSYSOPR, QPGMR, QSRV, and QSRVBAS user profiles unless the user has been granted specific authority to the command.

System/36 Environment

Command	Referenced Object	Authority Needed
CHGS36	Database file QSYSENV	Change
	Library #LIBRARY	
CHGS36PGMA	Program	Change
	Library	Read
CHGS36PRCA	File QS36SRC	Change
	Library	Read
CHGS36SRCA	Source	Use, management, and update
	Library for source	Read
	Change attributes	Update
CRTMSGFMNU	Display file if it already exists	All
	Message file	Use
	Library	Change
	Source file QS36SRC	All
CRTS36CBL	Library #COBLIB	Use
	Message files #CB#M1 and #CB#M2	Use
	Library QSBLMSG	Use
CRTS36DSPF	To-file source file when TOMBR is not *NONE	All
	To-file library	Change
	Source file QS36SRC	Use
	Display file when REPLACE(*YES) is specified	Operational and existence
	Display file library	Change
	Create Display File (CRTDSPF) command	Operational

System/36 Environment

Command	Referenced Object	Authority Needed
CRTS36MNU	To-file source file when TOMBR is not *NONE	All
	To-file library	Change
	Source file QS36SRC	Use
	Source file library	Read
	Display file when REPLACE(*YES) is specified	All
	Message files named in source	All
	Menu library	Change
	Display file library	Change
	Create Message File (CRTMSGF) command	Operational and existence
	Add Message Description (ADDMSGD) command	Operational
	Create Display file (CRTDSPF) command	Operational
CRTS36MSGF	To-file source file when TOMBR is not *NONE	All
	To-file library	Change
	Source file QS36SRC	Use
	Display file when REPLACE(*YES) is specified	All
	Message file named in source	All
	Message file named in source when OPTION(*ADD) or OPTION(*CHANGE) is specified	Change
	Message files named in source when OPTION(*CREATE) is specified	All
	Message file library	Change
	Create Message File (CRTMSGF) command	Operational and existence
	Add Message Description (ADDMSGD) command	Operational
	Change Message Description (CHGMSGD) command when OPTION(*CHANGE) is specified	Operational
EDTS36PGMA	Program	Change and management
EDTS36PRCA	File QS36PRC	Change and management
EDTS36SRCA	Source file QS36SRC	Change and management
RSTS36F	From-file	Use
	To-file	All
	Device file or device description	Use
RSTS36FLR [1,2,3]	From-folder	Use
	To-folder	All
	Device file or device description	Use
RSTS36LIBM	From-file	Use
	To-file	Change
	To-library	Change
	Device file or device description	Use

System/36 Environment

Command	Referenced Object	Authority Needed
SAVS36F	From-file	Use
	To-file	All
	Device file or device description	Use

Notes:

1 You need operational, management, existence, and all data authorities to the document if replacing it. You need operational and all the data authorities to the folder if restoring new information into the folders, or save system (*SAVSYS) special authority.

2 If used for a data dictionary, the only the authority to the command is required.

3 You must be enrolled in Office if the source folder is a document folder.

Tables

Command	Referenced Object	Authority Needed
CRTTBL	Library	Read and add
DLTTBL	Table	Existence
	Library	Read
WRKTBL	Table	Operational
	Library	Read

TCP/IP Commands

Command	Referenced Object	Authority Needed
ADDTCPLNK	Line description	Use
ADDTCPPORT		
ADDTCPRSI		
ADDTCPRTE		
CFGTCP		
CHGTCPA		
CHGTCPLNK	Line description	Use
CHGTCPRTE		
ENDTCPCNN		
ENDTCPLNK		
RMVTCPLNK		
RMVTCPPORT		
RMVTCPRSI		
STRTCPFTP	Table objects	Use
STRTCPTELN	Table objects	Use
VFYTCPCNN		
WRKNAMSMTP		
WRKTCPSTS		

Upgrade Order Information Data

Command	Referenced Object	Authority Needed
RMVACTTRA [1]	Program QLMADRMV	Use
WRKORDINF [2]	Program QMAWKORI	Use

Note:

1 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QSRV, and QSECOFR.

2 The public is not authorized to this command. The IBM-supplied user profiles that have authority to use this command are QPGMR, QSRV, and QSECOFR.

User Index

Command	Referenced Object	Authority Needed
DLTUSRIDX	Library for user index	Read
	User index	Existence

User Profiles

Command	Referenced Object	Authority Needed
CHGDSTPWD [1]		
CHGPRF[9]	User profile	Use and management
CHGUSRPRF [2,9,10]	User profile	Use and management
CRTUSRPRF [2,9,10]		
DLTUSRPRF [2]	User profile	Use and existence
DSPAUTUSR [3,4,11]		
DSPPGMADP [5,11]	User profile	Management
DSPUSRPRF[11]	User profile	Read
GRTUSRAUT [6,7]	Referenced user profile	Read
RSTAUT [4,8]		
RSTUSRPRF [4,8]		
RTVUSRPRF	User profile	Read

User Profiles

Command	Referenced Object	Authority Needed
WRKUSRPRF	User profile	Read

Notes:

1 Only the security officer (QSECOFR) user profile can use this command.

2 Security administrator (*SECADM) special authority is required.

3 This command is authorized to the security officer (QSECOFR) user profile or to a user with all object (*ALLOBJ) special authority specified in the user profile.

4 This command is restricted and shipped with authority for only the security officer. Other users must be specifically given authority to use it.

5 You must have *ALLOBJ special authority to display the IBM-supplied user profiles.

6 For files, libraries, and subsystem descriptions, operational authority is required.

7 You must be the owner or have management authority, and you need the authorities being granted.

8 You must have save system (*SAVSYS) special authority in your user profile to use this command.

9 You must have use authority to the current library, initial program, initial menu, job description, message queue, output queue, and attention-handling program. You must also have read authority to the library in which these objects exist.

10 You must have change and management authority to the group profile. Management authority cannot come from a program that adopts authority.

11 If OUTPUT(*OUTFILE) is specified, then the authority required by CLRPFM and ADDPFM commands is also required.

User Queue

Command	Referenced Object	Authority Needed
DLTUSRQ	Library for user queue	Read
	User queue	Existence

User Space

Command	Referenced Object	Authority Needed
DLTUSRSPC	Library for user space	Read
	User space	Existence

Utilities

Command	Referenced Object	Authority Needed
STRSDA	Source file	Read, add, update, and delete
	Update and add new member	Change and management
	To-library	Read and add
	Delete member	All
	Library for member	Read

Utilities

Command	Referenced Object	Authority Needed
STRSEU	Source file	Read, add, update, and delete
	Edit, add, or change a member	Operational and management
	Browse a member	Operational
	Print a member	Operational
	Remove a member	Operational and existence
	Library to add member	Read and add
	Change type or text of member	Operational
WRKLIBPDM [1]	Library	Read
WRKOBJPDM [1]	File	Read
	Library	Read

Note:

1 To use the individual operations, you must have the authority required by the individual operation.

Writers

Command	Referenced Object	Authority Needed
CHGWTR [1,2,3,4]	Output queue	Read, add, and delete
	Library for output queue	Read
ENDWTR [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
HLDWTR [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
RLSWTR [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
STRDKTWTR [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
	Message queue	Operational and add
	Device description	Read
STRPRTWTR [1,2,3]	Output queue	Read, add, and delete
	Library for output queue	Read
	Message queue	Operational and add
	Device description	Read

Notes:

1 If you have job control (*JOBCTL) special authority and OPRCTL(*YES) is specified for the queue, you do not need the authority specified by the AUTCHK parameter.

2 If you have spool control (*SPLCTL) special authority, you do not need the authority specified by the AUTCHK parameter.

3 Read, add, and delete authorities are used if the queue was created with AUTCHK(*DTAAUT) specified. You must be the owner of the queue if the queue was created with AUTCHK(*OWNER) specified.

4 To change the output queue for the writer, you must have 1, 2, or 3 to the new output queue.

Appendix E. Supported Call Level Interfaces

Table E-1 shows the call level interfaces, their shipped default public authority, their command equivalents (if any) and a brief description of their function.

The call level interfaces listed in the table are the only interfaces that IBM allows user-created programs to make calls to.

Table E-1 (Page 1 of 4). Call Level Interfaces

Program	Default Public Authority	Similar Commands	Description
QSNDDTAQ	*USE	None	Send a message to a data queue
QRCVDTAQ	*USE	None	Receive a message from a data queue
QMHQRDQD	*USE	None	Receive data queue description
QCLRDTAQ	*USE	None	Clear data queue
QCMD	*USE	None	Present the Command Entry display
QCL	*USE	None	Present the Command Entry display
QCMDEXEC	*USE	None	Run a command
QCAEXEC	*USE	None	Run a command in System/38 environment
QCACHECK	*USE	None	Validity check a command
QCMDCHK	*USE	None	Validity check a command
QCLSCAN	*USE	None	Scan for a string
QTBXLATE	*USE	None	Translate a character string
QDCXLATE	*USE	None	Translate a character string
QPGMMENU	*EXCLUDE	None	Show the programmer's menu
QPRCRTPG	*EXCLUDE	None	Create program
QUSLOBJ	*USE	DSPOBJD	List objects
QUSROBJD	*USE	RTVOBJD	Retrieve object description
QUSLMBR	*USE	DSPFD	List database members
QUSRMBRD	*USE	DSPFFD	Retrieve file member description
QUSLRCD	*USE	DSPFD, DSPFFD	List file record formats
QUSLFLD	*USE	DSPFFD	List file field descriptions
QUSLJOB	*USE	WRKSBSJOB	List jobs
QUSRJOBI	*USE	DSPJOB, WRKACTJOB	Retrieve job information

Table E-1 (Page 2 of 4). Call Level Interfaces

Program	Default Public Authority	Similar Commands	Description
QWCLJOBI (For Version 2 Release 1.1)	*USE	DSPJOB	List job information
QWCLOBJL (For Version 2 Release 1.1)	*USE	WRKOBJLCK	List object locks
QWCLASBS	*USE	WRKSBS	List active subsystems
QWDRSBSD	*USE	DSPSBSD	Retrieve subsystem information
QWDLSJBQ	*USE	DSPSBSD	List job queues for a subsystem
QUSLSPL	*USE	WRKSPLF, WRKOUTQ	List spooled files
QUSRSPLA	*USE	WRKSPLFA	Retrieve spooled file attributes
QUSCHGPA	*USE	CHGSBSD, CHGSYSVAL	Change pool attributes
QUSCMDLN	*USE	CALL QCMD	Display pop-up window command line
QUSCHGUS	*USE	None	Change user space
QUSCRTUI	*USE	None	Create user index
QUSCRTUQ	*USE	None	Create user queue
QUSCRTUS	*USE	None	Create user space
QUSDLTUS	*USE	DLTUSRSPC	Delete user space
QUSDLTUI	*USE	DLTUSRIDX	Delete user index
QUSDLTUQ	*USE	DLTUSRQ	Delete user queue
QUSPTRUS	*USE	None	Pointer to user space
QUSRTVUS	*USE	None	Retrieve user space
QREXX	*USE	STRREXPRC	Start the REXX interpreter to run REXX procedure.
QREXVAR	*USE	None	Manipulate variables in an actively running REXX procedure.
QREXQ	*USE	None	Manipulate entries on a job's REXX external data queue.
QWSQRYWS	*USE	None	Query device type-ahead setting.
QWSSETWS	*USE	None	Set device type-ahead setting.
QUHDSPH	*USE	None	Display help information
QWCCVTDT	*USE	None	Convert data and time
QWSSETWS	*USE	None	Control type ahead feature and keyboard buffering for display
QWSQRYWS	*USE	None	Query the current value for keyboard buffering for a display
QALGENA	*USE	None	Generate alert

Table E-1 (Page 3 of 4). Call Level Interfaces

Program	Default Public Authority	Similar Commands	Description
QALSNDA	*USE	None	Send an alert
QTVOPNVT	*USE	None	Open virtual terminal path
QTVRDVT	*USE	None	Read data from virtual terminal
QTVWRTVT	*USE	None	Write data to virtual terminal
QTVSNDRQ	*USE	None	Send request to the operating system
QTVCLOVT	*USE	None	Close virtual terminal path
QSYGETPH	*EXCLUDE	None	Get profile handle
QWTSETP	*EXCLUDE	None	Set profile
QSYRLSPH	*EXCLUDE	None	Release profile handle
QHFLSTFS	*USE	DSPHFS	List all file systems currently registered
QHFCTLFS	*USE	None	Transmit commands to your file system
QHFCRTDR	*USE	None	Create directory
QHFRNMDR	*USE	None	Rename directory
QHFDLTDR	*USE	None	Delete directory
QHFCPYSF	*USE	None	Copy stream file
QHFMOVSF	*USE	None	Move stream file
QHFRNMSF	*USE	None	Rename stream file
QHFDLTSF	*USE	None	Delete stream file
QHFOPNSF	*USE	None	Open stream file
QHFRDSF	*USE	None	Read from stream file
QHFWRTSF	*USE	None	Write to stream file
QHFLULSF	*USE	None	Lock and unlock range in stream file
QHFCHGFP	*USE	None	Change file pointer
QHFFRCSF	*USE	None	Force buffered data
QHFGETSZ	*USE	None	Get stream file size
QHFSETSZ	*USE	None	Set stream file size
QHFCLOSF	*USE	None	Close stream file
QHFOPNDR	*USE	None	Open directory
QHFRDDR	*USE	None	Read directory entries
QHFRTVAT	*USE	None	Retrieve directory entry attributes
QHFCHGAT	*USE	None	Change directory entry attributes
QHFCLODR	*USE	None	Close directory
QHFRGFS	*EXCLUDE	None	Register file system

Table E-1 (Page 4 of 4). Call Level Interfaces

Program	Default Public Authority	Similar Commands	Description
QHFDRGFS	*EXCLUDE	None	Remove registration from file system
QOLELINK	*USE	None	Enable a link for input/output on a communications line
QOLDLINK	*USE	None	Disable a communications line
QOLSETF	*USE	None	Activate/deactivate a filter for a communications link
QOLSEND	*USE	None	Perform output on a communications link
QOLRECV	*USE	None	Perform input on a communications link
QOLQLIND	*USE	None	Query an existing token-ring, Ethernet, or X.25 line description
QOLTIMER	*USE	None	Set or cancel timer.

Appendix F. Planning Forms for Security

This appendix has planning forms for user profiles, resources, and authorization lists. These forms can be removed and copied.

Resource Security Form

Object _____ Owner _____

Library name _____ Public authority _____

Object type _____

Authority holder Y, N _____

Object secured by authorization list _____

System-Defined Authority

*CHANGE - Change (*OBJOPR, *READ, *ADD, *UPD, *DLT)
*ALL - All (*OBJOPR, *OBJMGT, *OBJEXIST, *READ, *ADD, *UPD, *DLT)
*USE - Use (*OBJOPR, *READ)
*EXCLUDE - Exclude - No authority
*AUTLMGT - Authorization list management

User-Defined Authority

*OBJOPR - Object operational
*OBJMGT - Object management
*OBJEXIST - Object existence
*READ - Read
*ADD - Add
*UPD - Update
*DLT - Delete

User	Group or authorization list name	Authority	User	Group or authorization list name	Authority

Note: You may copy as necessary.

RV2L058-0

User Profile Form (Part 1)

Name _____
Position _____
Responsibilities _____

Use profile (Yes, No) _____
Group profile (Yes, No) _____
Group member (Yes, No) _____
Group profile name _____

The default values are in parentheses.
Required - *

*User _____
Password (*USRPRF) _____
Set password to expire (*NO) _____
Profile status (*ENABLED) _____
User class (*USER) _____
Assistance Level (*SYSVAL) _____

Current library (*CRTDFT) _____
Initial program *NONE) _____
Initial program (*NONE) _____
Library name (*LIBL) _____
Initial menu (MAIN) _____
Library name (*LIBL) _____
Limited capability (*NO) _____
Text (*BLANK) _____

Additional Parameters:

Special authority (*USRCLS) _____
Special environment (*SYSVAL) _____
Display sign-on information (*SYSVAL) _____
Password expiration interval (*SYSVAL) _____
Limit device sessions (*SYSVAL) _____
Keyboard buffering (*SYSVAL) _____
Maximum storage (*NOMAX) _____
Priority limit (3) _____
Job description (QDFTJOBD) _____
Library name (*LIBL) _____

Group profile (*NONE) _____
Owner (*USRPRF) _____
Group authority (*NONE) _____
Accounting code (*BLANK) _____
Document password (*NONE) _____
Message queue (*USRPRF) _____
Library name (*LIBL) _____
Delivery (*NOTIFY) _____
Severity (00) _____
Print device (*WRKSTN) _____

Output queue (*WRKSTN) _____
Library name (*LIBL) _____
Attn-key-handling program (*SYSVAL) _____
Library name (*LIBL) _____
Language identifier (*SYSVAL) _____
Country identifier (*SYSVAL) _____
Coded character set identifier (*SYSVAL) _____
User options (*NONE) _____
Authority (*EXCLUDE) _____

For Group Profiles ONLY:

Member name	Member name	Member name	Member name

Note: You may copy as necessary.

RV2L052-1

User Profile Form (Part 2)
Resource Security

User profile name: _____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Object name _____
Object type _____ Library _____
Authority _____
Purpose _____
_____

Note: You may copy as necessary.

RV2L061-0

Authorization List Form (Part 1)

Authorization list name _____ **Owner** _____

Text _____ **Public Authority** _____

User	Object Authority *ALL *CHANGE *USE *EXCLUDE	Auth List Mgt	Object			Data			
			Opr	Mgt	Exist	Read	Add	Update	Delete

Note: You may copy as necessary.

RV2L059-0

Authorization List Form (Part 2)

Authorization list name _____ Owner _____

Text _____ Public Authority _____

User	Object Authority *ALL *CHANGE *USE *EXCLUDE	Auth List Mgt	Object			Data			
			Opr	Mgt	Exist	Read	Add	Update	Delete

RV2L060-1

Note: You may copy as necessary.

Bibliography

This section lists publications that provide additional information about topics described or referred to in this manual. The manuals in this section are listed with their full title and order number, but when referred to in text, a shortened version of the title is used.

Communications Security

- *Communications: Advanced Peer-to-Peer Networking Guide*, SC41-8188
  Short Title: *APPN Guide*

  This manual provides information about the APPN support provided by the AS/400 system. It also contains security considerations for communications.

- *Communications: Advanced Program-to-Program Communications Programmer's Guide*, SC41-8189
  Short Title: *APPC Programmer's Guide*

  This manual is a guide for developing application programs that use advanced program-to-program communications (APPC) and for defining the communications environment for APPC communications. It also contains security considerations for developing the application programs.

- *Communications: Finance Communications Programmer's Guide*, SC41-8099
  Short Title: *Finance Communications Programmer's Guide*

  This manual describes how finance support communicates with a controller and how to set up finance support. It provides information for writing application programs to communicate with applications on the finance controller. It also contains security considerations when using finance communications.

- *Communications: Intersystem Communications Function Programmer's Guide*, SC41-9590
  Short Title: *ICF Programmer's Guide*

  This manual provides information needed to write application programs that use the AS/400 communications and OS/400 intersystem communications function (OS/400—ICF). It also provides security considerations for developing the application programs for communications.

- *Communications: Remote Work Station Guide*, SC41-0002
  Short Title: *Remote Work Station Guide*

  This manual provides information to the user for setting up and using remote work station support; such as display station pass-through, distributed host command facility, and 3270 remote attachment.

- *Transmission Control Protocol/Internet Protocol Guide*, SC41-9875
  Short Title: *TCP/IP Guide*

  This manual provides information about how the AS/400 system carries out TCP/IP and how TCP/IP relates to other AS/400 communications protocols and the OfficeVision/400 licensed program.

OfficeVision/400 Security

- *Systems Application Architecture* OfficeVision/400*: Managing OfficeVision/400*, SC41-9627
  Short Title: *Managing OfficeVision/400**

  This manual provides information on how to manage the day-to-day activities of OfficeVision/400.

- *Systems Application Architecture* OfficeVision/400*: Using OfficeVision/400 Word Processing*, SC41-9618
  Short Title: *Using OfficeVision/400* Word Processing*

  This manual provides information on how to use the word processing functions of OfficeVision/400.

Operations

- *New User's Guide*, SC41-8211
  Short Title: *New User's Guide*

  This manual provides beginner information about how to sign on and off; send and receive messages; respond to keyboard error messages; use function keys; use display, command, and help information; and control and manage jobs.

- *System Operator's Guide*, SC41-8082
  Short Title: *Operator's Guide*

  This manual provides information about how to use the system unit control panel and console, send and receive messages, respond to error messages, start and stop the system, and do system tasks.

Application Programming Interface (API) for Security

- *System Programmer's Interface Reference*, SC41-8223
  Short Title: *System Programmer's Interface Reference*

  This manual contains information about APIs that will help you with security on your system. The security APIs and their functions are:

QSYGETPH Get Profile Handle. Validates a user ID and password, and creates an encryped abbreviation called a profile handle for that user profile.

QWTSETP Set Profile. Switches the job to run under a new profile.

QSYRLSPH Release Profile Handle. Deletes a profile handle.

Programming and Utility Security

- *Application Development Tools: Source Entry Utility User's Guide and Reference*, SC09-1338
  Short Title: *SEU User's Guide and Reference*

  This manual provides information about using the AS/400 Application Development Tools source entry utility (SEU) to create and edit source members.

- *Backup and Recovery Guide*, SC41-8079
  Short Title: *Backup and Recovery Guide*

  This manual provides information about the different media available to save and restore system data, as well as a description of how to record changes made to database files and how that information can be used for system recovery and activity report information. This manual describes saving the security information, using auxiliary storage pools, and checksum protection.

- *Programming: Control Language Programmer's Guide*, SC41-8077
  Short Title: *CL Programmer's Guide*

  This manual provides a wide-ranging discussion of the AS/400 programming topics.

- *Programming: Control Language Reference*, SC41-0030
  Short Title: *CL Reference*

  This manual provides a description of all the AS/400 control language (CL) and its OS/400 commands. Each command includes a syntax diagram, parameters, default values, keywords, and an example.

- *Cryptographic Support/400 User's Guide*, SC41-8080
  Short Title: *Cryptographic Support/400 User's Guide*

  This manual provides a description of the data security capabilities of the AS/400 Cryptographic Support. Cryptographic support is not a part of the operating system. You can order the cryptographic licensed program from the IBM Software Division.

- *Distributed Data Management Guide*, SC41-9600
  Short Title: *DDM Guide*

  This manual provides information about remote file processing. It describes how to define a remote file to OS/400 distributed data management (DDM), how to create a DDM file, what file utilities are supported through DDM, and the requirements of OS/400 DDM as related to other systems.

- *Systems Application Architecture* Structured Query Language/400 Reference*, SC41-9608
  Short Title: *SQL/400* Reference*

  This manual provides information that describes SQL/400 statements and their parameters.

Index

A

B

C

D

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

Special Characters

Readers' Comments

Application System/400™
Security Concepts and Planning
Version 2

Publication No. SC41-8083-00

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name _____ Address _____

Company or Organization _____ _____

Phone No. _____ _____

Readers' Comments
SC41-8083-00

IBM®

Cut or Fo
Along Lin

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

ATTN DEPT 245
IBM CORPORATION
3605 HWY 52 N
ROCHESTER MN 55901-7899

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

Program Number: 5738-SS1

Printed in Denmark by
J. H. Schultz Print a|s
Copenhagen

SC41-8083-00